Analysing Trajectories of Mobile Users: from Data Warehouses to Recommender Systems

Franco Maria Nardini[1], Salvatore Orlando[2,1], Raffaele Perego[1], Alessandra Raffaetà[2], Chiara Renso[1], and Claudio Silvestri[2]

[1] ISTI-CNR, Pisa, Italy
[2] DAIS - Università Ca' Foscari Venezia, Italy

Abstract. This chapter discusses a general framework for the analysis of trajectories of moving objects, designed around a Trajectory Data Warehouse (TDW). We argue that data warehouse technologies, combined with geographic visual analytics tools, can play an important role in granting very fast, accurate and understandable analysis of mobility data. We describe how in the last decade the TDW models have changed in order to provide the user with a more suitable model of the reality of interest and we also cope with the challenge of semantic trajectories. As a use case we illustrate how the framework can be instantiated for realizing a recommender system for tourists.

1 Introduction

Recent advances in mobile network devices, sensors, and positioning technologies enabled the tracking of large amounts of moving objects: vehicles, animals, vessels and in large part also humans. These technologies produce huge data streams of observations, which can be stored and used to reconstruct the original objects trajectories. These movement data represent a treasure in term of the potential applications that can benefit from their analysis. A special interesting aspect is the possibility of enriching the pure spatio-temporal data with suitable knowledge bases, to semantically annotate and transform such trajectories into more meaningful data.

In this chapter we survey a general framework for creating, analysing and exploiting (possibly semantically enriched) trajectories that we have been developing in the last decade.

The central idea is that of a *Trajectory Data Warehouse* (TDW), with spatial and temporal dimensions, which is populated, via a suitable ETL process starting from raw trajectory data (essentially, spatio-temporal points or *samples*). The TDW relies on a flexible *conceptual model* with associated *spatio-temporal dimensions* and *hierarchies*. More specifically, the spatial domain can be structured according to the application requirements, by exploiting hierarchies of regular grids (like in [11, 10]) or of regions with ad-hoc shapes [9]. While a hierarchy of regular grids can be used to analyse objects that can move freely in the space, hierarchies with ad-hoc shapes are useful for objects whose movements are constrained, such as objects that can only move along a road network (e.g., cars).

The TDW is provided with an interface that allows for *visual* OLAP operations for the analysis of aggregate trajectory data, by integrating OLAP tools with visual analytics [2]. This permits to overcome the limits of the usual OLAP user interfaces. In fact, the table based representation commonly adopted by OLAP tools makes it very difficult for the user to grasp the relationships between areas in the same neighbourhood, the evolution of spatial measures in time, or the correlations of different measures. Visualisation is crucial: it can be seen simultaneously as the output and end-product of a knowledge discovery cycle and the starting point for further, interactive and visual, analysis.

The TDW, as described above, suffices to study several quantitative properties of trajectories, such as speed, traveled distance, or presence. However, in order to analyze information concerning semantic aspects such as the kind of places visited, the goals of the trajectories, the performed activity, transportation means, a semantic enrichment of the trajectory data is necessary. We discuss how semantic trajectories can be constructed from the original collected samples by properly combining movement data with suitable knowledge bases. We describe how the conceptual model of the TDW has to be modified to implement a *Semantic Trajectory Data Warehouse* allowing us to analyse semantic trajectories according to the above mentioned semantic dimensions.

As a use case for semantic analysis of trajectories we discuss a touristic recommendation system. We illustrate how trajectories, which have been semantically enriched, can be used to recommend personalized tours. Specifically, we outline the whole process, starting from the selection of a set of tourist trajectories, the enrichment step for properly transforming and enriching the trajectories for our purposes, and finally how the obtained semantic trajectories are exploited to suggest personalized sightseeing tours, by modeling and maximizing user interest and visiting time-budget.

Fig. 1 summarizes the overall (semantic) trajectory analysis framework discussed in this chapter. On the top of the figure we show the source data, i.e., a set of trajectory samples represented as small colored circles. Samples for the same trajectory are filled with the same color and connected by a gray dashed line that represents the actual movement of the object. These samples are fed to a module in charge of reconstructing the trajectories followed by the moving objects, possibly using a map of the visited geographic region. Once reconstructed, the trajectories can be processed directly by an ETL module to populate a TDW, which allows us to perform visual OLAP analyses (Section 3). Alternatively, we can exploit some knowledge bases, for example a set of categorized geographic Points of Interest (PoIs), to semantically enrich and eventually transform trajectories. In this last case, the ETL module is specialized to populate a semantic TDW, with a suitably extended conceptual model (Section 4). The overall framework of Fig. 1 can be instantiated for particular applications. In Section 5 we illustrate how it can be used for building a recommender system for tourists.

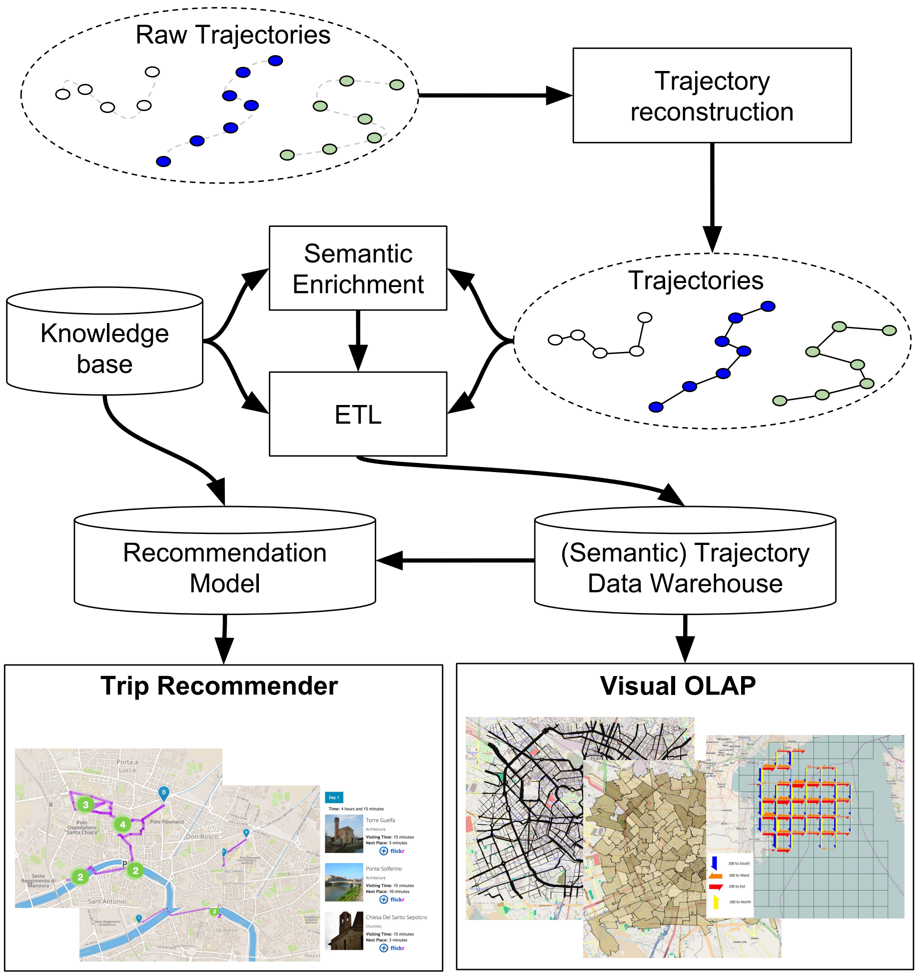


Fig. 1. Overview of the (semantic) trajectory analysis framework

2 Preliminaries

Several works in the literature address the analysis of trajectory data [15]. Even for the definition of a trajectory several variants exist, formalizing the general idea of a trajectory as a representation of the spatio-temporal evolution of a moving object. Since trajectories are usually collected by means of position-enabled devices, the notion of trajectory has to deal with the concept of *sampling* that is the action of the device to detect spatial and temporal points at given temporal intervals. Here we call *raw trajectory* the discrete representation of a trajectory as a sequence of spatio-temporal points or *samples* as collected by the device.

Definition 1 (Raw Trajectory). *A trajectory T is an ordered list of spatio-temporal points or* samples $p_1, p_2, p_3, \ldots, p_n$*. Each p_i is a tuple (id, x_i, y_i, t_i) where id is the identifier of a trajectory, x_i, y_i are the geographical coordinates of the sampled point, and t_i is the timestamp in which the point has been collected, with $t_1 < t_2 < t_3 < \ldots < t_n$.*

From these sampled data, according to the application requirements, we need to reconstruct the approximation of the real trajectory, modeled as a continuous function from time to geographic coordinates.

The possible methods for reconstruction are different, and depend on the scenarios on which we focus on. Objects can move almost freely in the space (e.g., vessels on the sea), or object movements can be constrained (e.g., cars moving along a road network). In the first case, in order to reconstruct the whole trajectory, *local interpolation* can be used. According to this method, objects are assumed to move between the observed points following some rule. For instance, a *linear* interpolation function models a straight movement with constant speed, while other polynomial interpolations can represent smooth changes of direction. If we consider the alternative scenario of cars moving along a road network, in turn modeled as a graph embedded in the Euclidean 2D-space, we have that the movements of objects are completely constrained, since cars are supposed to stay on the network. So reconstruction must take into account the topology of the road network to determine the path followed by each object between two consecutive sampled positions in the raw data [4]. The reconstruction phase produces a sequence of lines in a spatio-temporal space, each representing the continuous "development" of the moving object during a time interval. Notice that the spatial projection of these lines are segments of the road network or portions of these segments.

3 Trajectory Data Warehouses

The motivation behind a TDW is to transform trajectories into valuable knowledge that can be used for decision making purposes in ubiquitous applications, such as Location-Based Services (LBS) or traffic control management. Intuitively, the high volume of raw data produced by sensing and positioning technologies, the complex nature of data stored in trajectory databases and the specialized query processing demands, they all make extracting valuable information from such spatio-temporal data a challenging task. For this reason, the idea is to develop specific aggregation techniques to produce summarized trajectory information and provide *visual* OLAP style analyses.

3.1 The conceptual model

Our first proposal [11] of TDW consists of a fact table containing keys to dimension tables and a number of measures expressing properties about sets of trajectories. The dimensions of analysis are the spatial dimensions X, Y ranging over spatial intervals, and the temporal dimension T ranging over temporal intervals. A regular three-dimensional grid obtained by discretizing the corresponding values of the dimensions is defined and a set-grouping hierarchy is associated with each dimension. The measures of interest are the number of trajectories inside each cell of the grids, their average, maximum/minimum speed, the covered distance and the time spent inside the cell. Then in [14] we add a new dimension *OBJECT_PROFILE_DIM* in order to take into account demographical information, such as gender, age, job, of moving objects. However, these approaches suffer from a main limitation: they are restricted to freely moving objects. Thus, they do not allow to explicitly account for constrained movements, for example due to the presence of a road network. Moreover, they support only spatitro-temporal hierarchies consisting of regular grids.

In [9] we define a framework relying on a more flexible conceptual model with associated spatio-temporal dimensions and hierarchies. More specifically, the spatial domain can be structured according to the application requirements, by exploiting hierarchies of regular grids (like in [11, 14]) or of regions with ad-hoc shapes. While a hierarchy of regular grids can be used to analyse objects that can move freely in the space, hierarchies with ad-hoc shapes are useful for objects whose movements are constrained, such as objects that can only move along a road network (e.g., cars). Furthermore, Voronoi tessellation can be employed in order to build hierarchies of regions based on the actual distribution of the points forming the trajectories. This kind of partitioning turns out to be particularly suited for highlighting the directions of the trajectory movement.

The resulting model is presented in Fig. 2. We distinguish two classes of facts, namely INTRA-GRANULE and INTER-GRANULE facts. Intuitively, intra-granule facts express properties related to trajectories inside a *single* granule whereas inter-granule facts describe properties concerning the movement of trajectories between two granules. We recall that a *base granule* is obtained by partitioning both the spatial and temporal dimensions and this partition is the finest one. From this base granularity other *coarser* partitions can be defined by merging together spatial regions and temporal intervals, respectively. Informally, a granule can be defined as a contiguous spatial region during a given time interval.

More specifically, the INTRA-GRANULE facts model events that are related to a *single base granule* concerning a certain object group. For a given object group U and a granule g, the measures are:

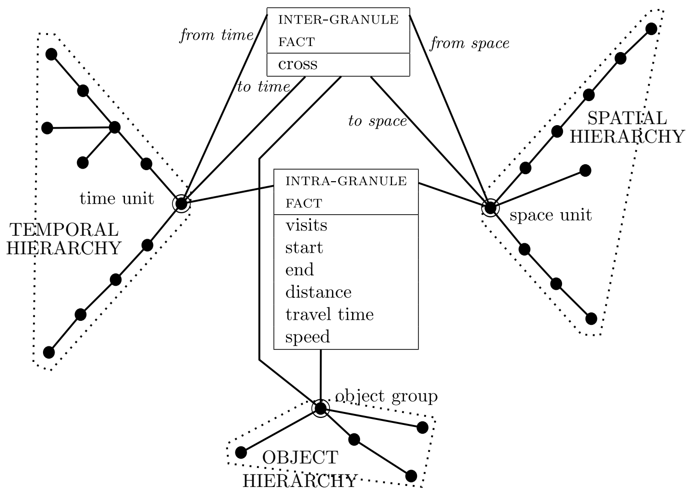


Fig. 2. TDW Conceptual model [9]

- *visits*: the number of trajectories belonging to group U which start from or enter into granule g;
- *start/end*: the number of trajectories belonging to group U starting/ending in granule g;
- *travel time/distance*: the time spent/distance travelled by all trajectories belonging to group U while moving inside granule g;
- *speed*: the average speed of trajectories belonging to group U traversing granule g.

The INTER-GRANULE facts model events that are related to *pairs of granules* and are concerned with a specific object group. For a given group U and pair of granules g and g', a measure of interest is

- *cross*: number of times the border from g to g' has been traversed by trajectories belonging to group U.

Note that the measure *cross* is interesting only for adjacent granules (for non-adjacent granules it is invariably 0). However, in general, inter-granule facts can model events which are meaningful for all pairs of granules. An example could be the *origin-destination* measure, which, for any pair of granules, represents the number of trajectories starting from the first and ending into the second granule.

Clearly, the presented measures are not an exhaustive collection, but they correspond to a set of common measures which we found interesting and useful in different scenarios.

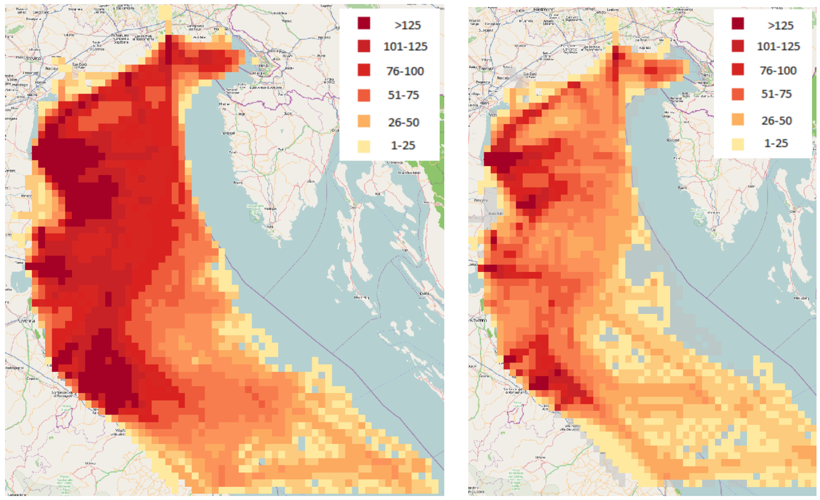
The TDW provides efficient OLAP roll-up operations since for all the defined measures, values at a coarser granularity can be computed by using values at a finer granularity. In particular, for the measures *start*, *end*, *travel time*, *distance* and *cross*, we use the *distributive* function *sum* as aggregate function whereas for *visits* and *speed* we use *algebraic* aggregate functions. The aggregate function for *speed* is computed as the ratio between the measures *distance* and *travel time*, as expected. On the other hand, for the measure *visits* we use the auxiliary measure *cross*. To give an intuition, let us consider a granule g composed by two finer granules g_1 and g_2. Hence the number of visits in the granule g is obtained by summing up the visits in g_1 and g_2, subtracting the number of trajectories crossing the border between g_1 and g_2. This is motivated by the fact that the border between two finer granules, g_1 and g_2 composing g, is completely inside g. Hence trajectories moving from g_1 to g_2 (or vice versa) increase the number of visits in g_1 (or g_2) but they should not be counted as visits in the coarser granule g because the movement is completely inside g, i.e., they do not enter g. We refer the reader to [9] for a formal definition of such aggregate functions.

It is worth noting that measure *visits* can provide an accurate approximation of measure *presence*, which counts the number of *distinct* trajectories occurring in a spatio-temporal granule. The aggregate function for *presence* is *holistic*: the raw data are needed to compute the exact result at all granularities. This is due to the fact that trajectories might span multiple granules. Hence in the aggregation phase we have to cope with the so called *distinct count problem* [17]: if an object remains in the query region for several timestamps during the query interval, one should avoid to count it multiple times in the result. *Holistic* functions represent a big issue for data warehouse technology. In [9] we discussed about the computation of measure *presence* and we showed that the proposed solution, i.e., the use of measure *visits*, is a more precise approximation with respect to some common approaches [17, 12] facing the same problem.

3.2 Visual OLAP

In the analysis of spatial and spatio-temporal data, the use of suitable, interactive, visualization tools is of paramount importance to help the analytic user in effectively grasping the information hidden in those complex data. For this reason, we have provided the TDW with an interface that allows for OLAP visual operations, based on V-Analytics [1, 2], an interactive visual analytics system running on the Java® Virtual Machine. This system permits a user to view georeferenced data over a map and run analyses on them, for example to find clusters or to tessellate the space. It also offers functionalities to handle temporal data, by using graphs or animations, according to the type of data to analyse.

In the following we report some examples that highlight several kinds of OLAP analyses on different scenarios. A first one, illustrated in Fig. 3(a), shows the fishing effort index in the Northern Adriatic Sea during 2007, at the base spatial granularity. The fishing effort index is a value indicating how much a given area has been exploited by the boats fishing in it. The space is partitioned into a regular grid and granules in darker colours are the most exploited. By using a drill-down operation on the temporal dimension, we can inspect the situation at a higher level of detail. For instance, Fig. 3(b) shows the fishing effort in the trimester July-September of 2007. The fact that it is sensibly reduced with



(a) January-September 2007 (b) July-September 2007

Fig. 3. Fishing effort distribution [9].

respect to effort in the whole period is somehow expected due to a law which prevents most fishing activities during August.

The flexibility in the definition of the spatio-temporal hierarchy offered by the presented TDW model allows the user to adopt a suitable model of the reality, thus obtaining a much more meaningful visual representation of the information contained in the TDW. The images in Fig. 4 are relative to a different example that concerns trajectories of cars moving in the city of Milan. Specifically, they visually represent the number of visits to spatial granules during the time interval corresponding to a particular temporal granule. Each image corresponds to a different spatial granularity: in Fig. 4(a) granules are cells of a regular grid, whereas in Fig. 4(b) and in Fig. 4(c) granules are respectively street segments and city districts. The results obtained with a regular grid may be suited for getting an initial overview of the data. However, a more detailed exploration is complicated since the cells do not bear any semantics and do not correspond to the real geographic and topographic properties of the data. This is why it is important to have also streets and district for the analyses. For example, by using the streets we can detect which are the most busy roads and how the traffic flows.

4 Semantic Trajectory Data Warehouse

The concept of semantic trajectory has been proposed as a way to overcome the lack of semantics characterizing raw trajectories. A well known definition of semantic trajectory relies on the "stop and move" approach: a trajectory is segmented into parts where the object is stopped (the "stop") and the parts where

(a) (b) (c)

Fig. 4. (a) Grid based spatial dimension, and (b) street segment based spatial dimension with (c) dimensional attribute having polygon spatial type. [9]

the object is changing his/her position (the "move") [16]. This approach evolved to the more general definition of *episodes* to represent segments of a trajectory complying to some predicate representing the semantics of that segment, like the transportation mean, the goal or activity [13]. A further evolution towards this direction brought to the definition of a conceptual model for semantic trajectories as proposed in [3] where several contextual aspects contribute to create the concept of semantic trajectory.

Definition 2 (Semantic Trajectory). *A semantic trajectory is a trajectory that has been enhanced with annotations and/or one or several complementary segmentations.*

Note that, according to the specific requirements of applications, such semantic trajectories can be transformed and abstracted so to adhere to a model, e.g., the "stop and move" one. Therefore, while semantics enrichment can add meaningful information to trajectories, the obtained semantic trajectory can actually lose some of the information contained in the original one, by keeping only that useful for the specific application goal.

4.1 The Semantic TDW conceptual model

This section introduces the Mob-Warehouse model [18] which is organized around the notion of semantic trajectory where different aspects contribute to describe the context. The model is based on the so called *5W1H* (Who, Where, When, What, Why, How) framework [19], recurrently used by journalists as a guide for narrating a fact.

To semantically enrich a trajectory, each narrative question of the 5W1H model is mapped to a specific trajectory feature. In this way, we describe the moving object (*Who*) moving by a transportation means and/or having a certain behavior (*How*), performing an activity (*What*), for a certain reason (*Why*), at a given time (*When*) and place (*Where*). The increased level of semantic information into our model trajectory allows us to perform more meaningful queries about moving object habits.

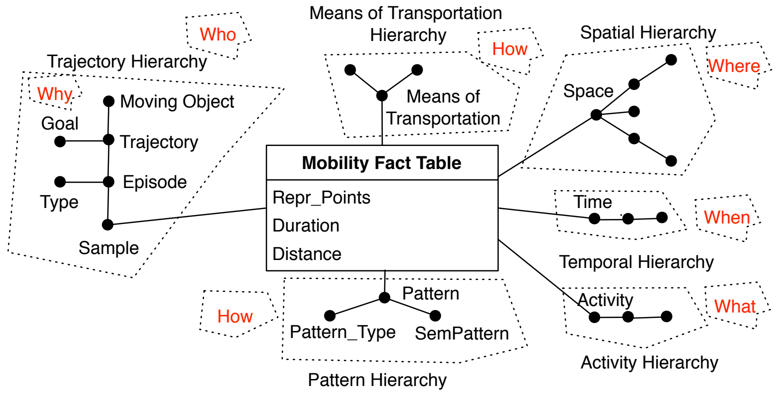


Fig. 5. Semantic TDW Conceptual model [18]

The resulting semantic TDW conceptual model consists of six dimensions, as illustrated in Fig. 5. It reflects the *semantic* structure of trajectories and in the fact table we store detailed information, no more aggregate properties about trajectories as in the model presented in Section 3. Specifically, the dimensions *Space* and *Time* essentially correspond to the previous model dimensions and they represent respectively the *Where* and *When* questions of the 5W1H Model. It is worth noting that the spatial domain can be structured according to the application requirements, providing the user with a great flexibility. The third dimension named *object group* in Fig. 2 representing features of the objects under analysis, here it is called dimension *Trajectory*, as it becomes a central component of our model and it is used to represent the trajectory of the moving objects. At the base granularity it represents a single sample (id, x, y, t) belonging to the trajectory identified by *id*. The hierarchy having *Sample* as a root mixes together semantic and geometric features. A sample belongs to an *episode*, which can be classified according to its *Type* (e.g., a *stop* or a *move*) and it is grouped into a *Trajectory*. Each *Trajectory* is associated not only with the *Moving object* but also to a *Goal*, which is the main objective of such a trajectory. This dimension allows one to model *Who* is performing the action (the moving object) and the attribute *goal* answers the question *Why*. A fourth dimension, called *Activity*, states the activity the object is doing in a certain sample. This allows one to describe in a very detailed manner *What* is going on at the different samples of a trajectory. We can build a hierarchy of activities which classifies properly the variety of tasks an object can perform. Usually this hierarchy is application dependent hence in the general model it is not specified and should be instantiated case by case depending on the application requirements. Then the dimension *Means of Transportation* represents which transportation means the object is using for the movement. The last dimension, called *Pattern*, collects the patterns mined from the data under analysis. In this way we can directly relate trajectories to the patterns they belong to. The latter two dimensions express the concept of *How* the movement is performed.

The fact table stores measures about the samples of the trajectories. Unlike [10, 14, 9] where at the minimum granularity data were already aggregated, in this model we record the most detailed information. This gives the user the ability to analyze the behavior according to various points of view: at the minimum granularity we store information related to a single sample of a trajectory, specifying the kind of activity is doing, the means of transportation is using, the patterns, the space and time it belongs to. Then by aggregating according to the described hierarchies we can recover also properties concerning the whole trajectory or groups of trajectories satisfying certain conditions.

In the fact table we store measures related to a given sample $s = \langle id, x, y, t \rangle$

- *Repr_Points* is a spatio-temporal measure containing the spatial and temporal component of the sample, i.e., (id, x, y, t);
- *Duration* is the time spent to reach the sample from the previous point of the same trajectory in the same granule. It is set to 0, if this is the first point of the trajectory in such a granule;
- *Distance* is the traveled distance from the previous point to the sample of the trajectory in the same granule. It is set to 0, if this is the first point of the trajectory in such a granule.

As far as the aggregate functions are concerned, for the measures *Duration* and *Distance*, we use the *distributive* function *sum*: super-aggregates are computed by summing up the sub-aggregates at finer granularities. On the other hand, the aggregate function for the measure *Repr_points* can be defined in different ways according to the application requirements. The simplest way is to use the *union* operator to join together the points satisfying given conditions. Differently one can return a bounding box enclosing all the points or compress the points removing the ones which are spatio-temporally similar.

5 Use Case: Trip Planning Recommender for Tourists

Planning a travel itinerary is a difficult and time-consuming task for tourists approaching their destination for the first time. Different sources of information such as travel guides, maps, on-line institutional sites and travel blogs are consulted in order to devise the right blend of Points of Interest (PoIs) that best covers the subjectively interesting attractions and can be visited within the limited time planned for the travel. However, the user still need to guess how much time is needed to visit each single attraction, and to devise a *smart* strategy to schedule them moving from one attraction to the next one. Furthermore, tourist guides, and even blogs, reflect the point of view of their authors, and they may result to be not authoritative sources of information when the tourist preferences diverge from the most popular flow.

We show how, relying on our framework, we can build a personalized plan of visit by exploiting the wisdom-of-the-crowds by past tourists. First of all we have to select and/or create the *Knowledge bases* (see Fig. 1) that can be used both for the Semantic Enrichment and during the ETL phase. In order to suggest interesting itineraries, we have to identify the set of PoIs in the geographical region that tourists would like to visit. Given the bounding box BB_{city} containing the city of interest, we download all the geo-referenced Wikipedia pages falling within this region. We assume each geo-referenced Wikipedia named entity, whose geographical coordinates falls into BB_{city}, to be a fine-grained Point of Interest. For each PoI, we retrieve its descriptive label, its geographic coordinates as reported in the Wikipedia page, and the set of categories which the PoI belongs to. Categories are reported at the bottom of the Wikipedia page, and are used to link articles under a common topic. They form a hierarchy, although sub-categories may be a member of more than one category. By considering the set C of categories associated with all the PoIs, we generate the normalized relevance vector of each PoI.

We then perform a density-based clustering to group in a single PoI sightseeing entities which are very close one to each other[3]. Clustering very close PoIs is important since a tourist in a given place can enjoy all the attractions in the surroundings even if she does not take photos to all of them. Moreover, it aims at reducing the sparsity that might affect trajectory data. Finally, we obtain the relevance vector for the clustered PoIs by considering the occurrences of each category in the members of the clusters and by normalizing the resulting vector. The final result is a knowledge base consisting of a set of PoIs $\mathcal{P} = \{p_1, \ldots, p_N\}$ and each POI is associated with the *relevance vector* $\boldsymbol{v_p} \in [0, 1]^{|C|}$.

Now we need a method for collecting users $\mathcal{U}$ and the long-term itineraries crossing the discovered PoIs. We query Flickr to retrieve the metadata (user id, timestamp, tags, geographic coordinates, etc.) of the photos taken in the given area BB_{city}. The assumption we are making is that photo albums made by Flickr users implicitly represent sightseeing itineraries within the city. To strengthen the accuracy of our method, we retrieve only the photos having the highest geo-referenced accuracy given by Flickr[4]. This process thus collects a large set of geo-tagged photo albums taken by different users within BB_{city}. We preliminary discard photo albums containing only one photo and the resulting set represents the set of *Raw Trajectories* in Fig. 1.

Then, we apply a *Semantic Enrichment* step. We spatially match the photos in the raw trajectories against the set of PoIs previously collected. We associate a photo with a PoI when *it has been taken within a circular buffer of a given radius having the PoI as its center*. Note that in order to deal with clustered PoIs, we consider the distance of the photo from all constituent members: in the case the photo falls within the circular region of at least one of the members, it is assigned to the clustered PoI. Moreover, since several photos by the same user are usually taken close to the same PoI, we consider the timestamps associated with the first and last of these photos as the starting and ending time of the user visit to the PoI. At the end of this step we have a set of *semantic* trajectories

---

[3] E.g., the beautiful marble statues in the *Loggia dei Lanzi* in Florence are only a few meters far one from each other but have a distinct dedicated page in Wikipedia.

[4] `http://www.flickr.com/services/api/flickr.photos.search.html`

consisting of sequences of PoIs belonging to P and each PoI is annotated also with the time the user is assumed to enter and to exit from such a PoI.

It is worth noting that in this case semantic trajectories are sequences of *stops* since the selected dataset does not provide any information about the movements of the user from one stop to the following. For the purpose of our application, in the ETL phase the *moves* are computed as the shortest path between two consecutive stops by using Googlemaps. Moreover, the set of PoIs is further annotated with the *visiting time* and the *popularity* index. The visiting time for a PoI p is the time spent by users in p and it is computed as the average of the durations of the visits to p. The popularity of each PoI is computed as the number of distinct users that take at least one photo in its circular region. The set of PoIs is used to build the spatial dimension of the Semantic TDW.

Finally, it is possible to associate a *preference vector* with each user by summing up and normalizing the relevance vectors of all the PoIs occurring in the semantic trajectories of such a user.

The general Semantic TDW model of Fig. 5 when instantiated to this use case includes only the dimensions *Space*, *Time* and *Trajectory*, since the raw data do not provide any specific information on means of transportation, activities and patterns. It is important to highlight that this data warehouse can support analyses at different levels of abstraction: from very detailed data involving samples to semantic trajectories modeled as sequences of stops and moves.

The *Trip Planning Recommendation* [5–7] is an example of analysis that can be performed on top of the Semantic TDW. The aim is to generate visiting plans made up of actual touristic itineraries that are the most tailored to the specific preferences and the temporal constraints of the tourist. The Trip Planning Recommendation is defined as a set cover problem, formulated as an instance of the Generalised Maximum Coverage (GMC) problem [8]. We model each visiting pattern by means of the PoIs and the associated Wikipedia categories, and the GMC profit function by considering PoIs popularity and the actual user preferences over the same Wikipedia categories. The cost function is instead built by considering the average visiting time for the PoIs in the patterns plus the time needed to move from one PoI to the next one.

Given a tourist, the Trip Planning Recommendation problem can be thus solved by looking at the set of semantic trajectories fitting the available time budget and covering the PoIs, that maximises the user interests. Determining an exact solution for the Trip Planning Recommendation problem is NP-hard. We solve it by employing the efficient greedy approximation algorithm proposed in [8]. Trajectories are then scheduled and provided to the user as an agenda of activities to be performed in the city. An example of the recommendation produced for the city of Pisa is shown in Fig. 6.

6 Conclusion

This chapter surveyed some research results obtained by the authors in the field of trajectory data analysis. Our achievements are discussed by referring to a

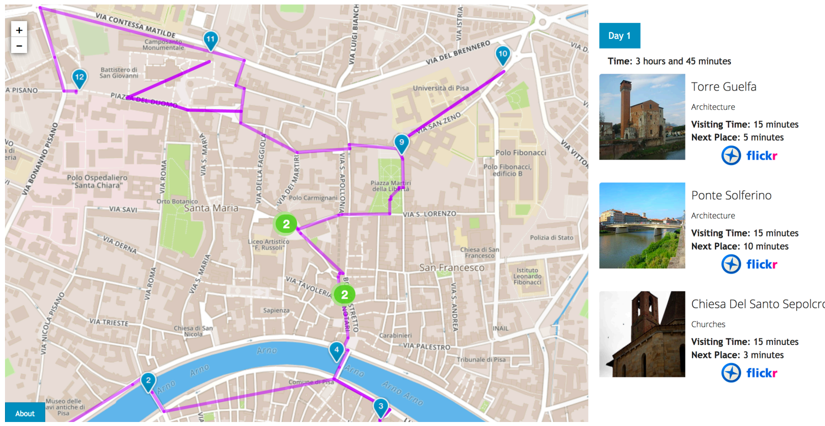


Fig. 6. An example of trip plan recommendation [5, 7, 6].

general framework that encompasses many steps, from semantic enrichment of trajectories, in turn reconstructed from sequences of samples, to Data Warehousing.

The central part of this chapter refers to a general conceptual model for TDW presented in [9], with associated spatio-temporal dimensions, where the spatial domain can be structured according to the application requirements and it is no longer restricted to consist of simple regular grids as in previous works. Moreover, the TDW is provided with a set of spatial and temporal visualisation techniques, supporting OLAP analysis of movement data, which permits a user to view geo-referenced data over a map and run insightful analyses on them.

An extension of the TDW model for semantically enriched trajectories is also discussed, by presenting the Mob-Warehouse model [18]. In this case, the most notable contribution is the semantic conceptual TDW model based on the so called *5W1H* (Who, Where, When, What, Why, How) framework. Each narrative question of the 5W1H model is mapped to a specific trajectory feature. In this way, we describe an object (Who) moving by a transportation means and/or having a certain behavior (How), performing an activity (What), for a certain reason (Why), at a given time (When) and place (Where).

We finally illustrated a trip planning recommender in the context of a tourism scenario [5–7]. We analyzed its various steps to eventually recommend a trip plan to a user, where her profile and time budget is known, on the basis of a recommendation model extracted from past trajectories of tourists.

References

1. G. Andrienko, N. Andrienko, and S. Wrobel. Visual Analytics Tools for Analysis of Movement Data. *ACM SIGKDD Explorations*, 9(2):28–46, 2007.
2. Natalia Andrienko and Gennady Andrienko. Visual analytics of movement: An overview of methods, tools and procedures. *Inf. Visualization*, 12(1):3–24, 2013.

3. V. Bogorny, C. Renso, A. R. de Aquino, F. de Lucca Siqueira, and L.O. Alvares. CONSTAnT – A Conceptual Data Model for Semantic Trajectories of Moving Objects. *Transactions in GIS*, 18(1):66–88, 2014.
4. S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk. On Map-Matching Vehicle Tracking Data. In *Proc. of VLDB*, pages 853–864, 2005.
5. I. Ramalho Brilhante, J. A. F. de Macêdo, F. M. Nardini, R. P., and C. Renso. On planning sightseeing tours with TripBuilder. *Information Processing & Management*, 51(2):1–15, 2015.
6. I. Ramalho Brilhante, J. A. F. de Macêdo, F. M. Nardini, R. Perego, and C. Renso. Where Shall We Go Today? Planning Touristic Tours with TripBuilder. In Qi He, Arun Iyengar, Wolfgang Nejdl, Jian Pei, and Rajeev Rastogi, editors, *Proc. of CIKM*, pages 757–762. ACM, 2013.
7. I. Ramalho Brilhante, J. A. F. de Macêdo, F. M. Nardini, R. Perego, and C. Renso. TripBuilder: A Tool for Recommending Sightseeing Tours. In *Proc. of ECIR*, pages 771–774, 2014.
8. R. Cohen and L. Katzir. The generalized maximum coverage problem. *Information Processing Letters*, 108(1):15–22, 2008.
9. L. Leonardi, S. Orlando, A. Raffaetà, A. Roncato, C. Silvestri, G. L. Andrienko, and N. V. Andrienko. A general framework for trajectory data warehousing and visual OLAP. *GeoInformatica*, 18(2):273–312, 2014.
10. G. Marketos, E. Frentzos, I. Ntoutsi, N. Pelekis, A. Raffaetà, and Y. Theodoridis. Building Real World Trajectory Warehouses. In *Proc. of MobiDE*, pages 8–15, 2008.
11. S. Orlando, R. Orsini, A. Raffaetà, A. Roncato, and C. Silvestri. Trajectory Data Warehouses: Design and Implementation Issues. *Journal of Computing Science and Engineering*, 1(2):240–261, 2007.
12. D. Papadias, Y. Tao, P. Kalnis, and J. Zhang. Indexing Spatio-Temporal Data Warehouses. In *Proc. of ICDE*, pages 166–175, 2002.
13. C. Parent, S. Spaccapietra, C. Renso, G. Andrienko, N. Andrienko, V. Bogorny, M. L. Damiani, A. Gkoulalas-Divanis, J. A. Macedo, N. Pelekis, Y. Theodoridis, and Z. Yan. Semantic Trajectories Modeling and Analysis. *ACM Computing Surveys*, 45(4), 2013.
14. A. Raffaetà, L. Leonardi, G. Marketos, G. Andrienko, N. Andrienko, E. Frentzos, N. Giatrakos, S. Orlando, N. Pelekis, A. Roncato, and C. Silvestri. Visual Mobility Analysis using T-Warehouse. *International Journal of Data Warehousing and Mining*, 7(1):1–23, 2011.
15. C. Renso, S. Spaccapietra, and E. Zimányi, editors. *Mobility Data: Modeling, Management, and Understanding.* Cambridge University Press, 2013.
16. S. Spaccapietra, C. Parent, M. L. Damiani, J. A. de Macedo, F. Porto, and C. Vangenot. A conceptual view on trajectories. *Data & Knowledge Engineering*, 65(1):126–146, 2008.
17. Y. Tao, G. Kollios, J. Considine, F. Li, and D. Papadias. Spatio-Temporal Aggregation Using Sketches. In *Proc. of ICDE*, pages 214–225, 2004.
18. R. Wagner, J. A. F. de Macêdo, A. Raffaetà, C. Renso, A. Roncato, and R. Trasarti. Mob-Warehouse: A Semantic Approach for Mobility Analysis with a Trajectory Data Warehouse. In *Proc. of SeCoGIS*, volume 8697 of *LNCS*, pages 127–136, 2013.
19. L. Yang, Z. Hu, J. Long, and T. Guo. 5W1H-based Conceptual Modeling Framework for Domain Ontology and Its Application on STPO. In *Proc. of SKG*, pages 203–206, 2011.