Visually–driven analysis of movement data by progressive clustering

Extended Abstract

S. Rinzivillo D. Pedreschi
KDD Lab, University of Pisa
{rinziv,padre}@di.unipi.it

M. Nanni F. Giannotti
KDD Lab, ISTI – CNR, Pisa
{Mirco.nanni,fosca.giannotti}@isti.cnr.it

N. Andrienko G. Andrienko
Fraunhofer Institute IAIS
{gennady.andrienko, natalia.andrienko}@iais.fraunhofer.de

Motivations

The large availability of movement data is posing the basis for the development of novel classes of applications [1]. When the traces of moving objects are sensed by some wireless network infrastructure and recorded in a database, it becomes difficult to comprehend the common movement behaviour of groups of objects, even in presence of a few hundreds of trajectories.

Given a dataset of very simple data objects, e.g. bi-dimensional points, it is possible to identify a group of points by seeing a plot of the data on a display. With a set of trajectories, i.e., sequences of time-stamped locations of some moving objects, this task becomes quite problematic. In fact, even with a very few trajectories, the display may result in a spaghetti-like drawing where it is difficult also to distinguish each trajectory from the others. Thus, a method to group objects together is essential for simplifying the visualization to the user. In this sense, clustering techniques may be of great use in exploratory analysis of movement data. However, the role of clustering is not limited just to simplifying the drawings on the screen. In combination with appropriate visualisation and interactive techniques, this is a powerful tool for analysis. An earlier paper [2] presents a general framework for analysis of movement data and briefly describes the use of cluster analysis as a part of the framework. In this paper, we focus in more detail on the use of cluster analysis with different similarity measures.

In the complex data domain of trajectories and movement data, it is difficult to define a similarity measure that takes into account all the attributes of the objects. It is often the case that a similarity measure just focuses on one or a very few dimensions of the data to analyze. Thus, it is straightforward to present to the user a set of similarity functions in order to give her the possibility of choosing the one that better fits with her analytical objectives. In particular, since we assume that each distance function aggregates objects according to its own semantics, the user can choose a sequence of functions to be used progressively. Intuitively, at the very first application of the clustering method a rough distance is used. Then, a refined distance is applied for each of the resulting clusters of the first step. In this way, at each step, the user has a vision of a particular cluster and the similarities that lead to its definition.

In this paper, we adopt a visual analytics perspective, i.e., we assume that the interactive geographic visualization of the trajectory clusters obtained at each step is the best means to assess the quality of such results, and to progressively drive further refinements of the underlying clustering engine towards comprehensible clusters, which provide useful answers to analytical questions – actually, we shall see how new analytical questions can arise as an outcome of previous analysis. In this extended abstract, we limit ourselves to illustrate by means of an example how a visually-driven clustering methodology may assist the analyst in the process of creating and answering complex queries. In the expanded version, we shall present in full details such a methodology, emphasizing the important role that the different similarity definitions play; we shall also briefly discuss how such a methodology can, in principle, scale to reasonably large mobility datasets so that interactive visual analytics is supported also in realistic situations.

Related Work

The literature about clustering methods is very large. However, in the last few years, the availability of moving objects databases has offered the opportunity of extendig such methods also for spatio-temporal data. In particular, in [3] it has been proposed a method to apply a density-based clustering algorithm, namely OPTICS [4][5], to the trajectory domain. The distance function used in the paper is based on the average Euclidean distance between each point of two trajectories. In [6] Pelekis et al. present a framework consisting of a set of distance operators based on primitive (space and time) as well as derived parameters of trajectories (speed and direction).

There are several proposals in the literature for the interactive clustering through cluster analysis to identify relevant text documents. These methods are based on the k-means algorithm and the user interaction is exploited for refining process consists of moving objects among the defined clusters in order to augment the quality of each partition.

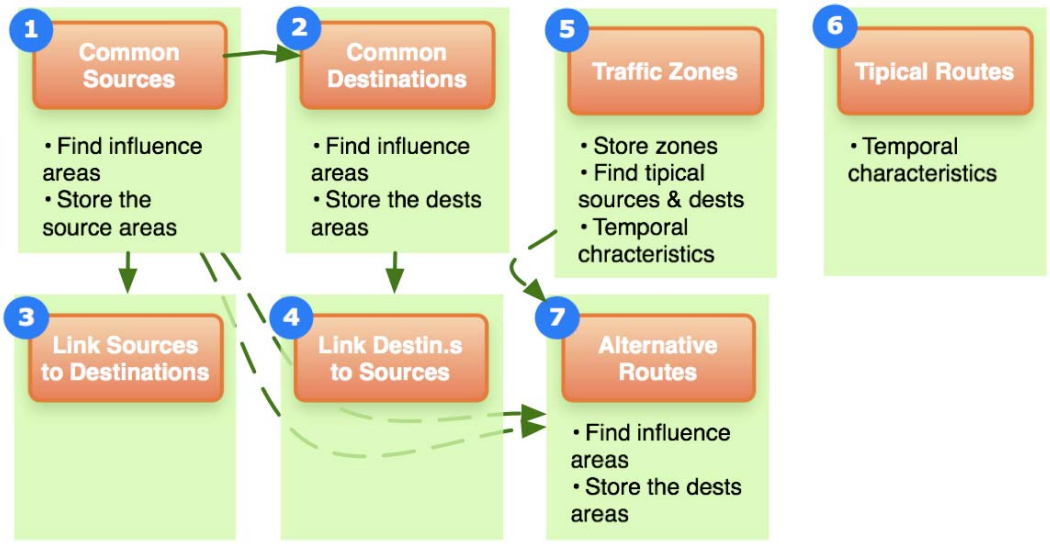
Some recent proposals focus on the visualization of multidimensional data for interactive cluster analysis [7][8]. In [7] the high-complexity of data is reduced by means of sampling or projection to present to the user a summary view of the whole dataset. The process of trajectory clustering should not be considered as a generalization of the clustering of n-dimensional data. The approaches presented in the literature that deal with multi-dimensional data present to the user a graphical representation of the dataset (or a sample of it) to allow her to select partitions or attributes to be used in the clustering . In a trajectory dataset the situation is quite different, since the evaluation of the clustering results should be related to the background maps and other background knowledge highlighted on the map. This pose the problem of finding an effective way of presenting to the user the clusters obtained from the mining algorithm in order to allow her to evaluate the implicit relations with the background.

Distance Functions

A distance function models the similarity of two trajectories by means of a mathematical function that compares the two entities and gives a weight to be used by the clustering algorithm. However, the definition of a one-for-all function is not simple. One may choose to compare the destinations of two trajectories, or their origin places, or their paths. In all these cases, the definition of unique distance function would produce a function that is both complex and time consuming. On the contrary, the definition of simpler functions results in more efficient distances. But, what if the analyst wants to group all the trajectories starting from the same location and following the same path?! Simply, she can apply the two distances in cascade, by selecting the groups of trajectories starting from a common area and then selecting, in each of the groups found, the trajectories following the same path. Besides its intuitiveness, this approach improves the scalability of the whole clustering process, since each distance function can be optimized by exploiting its own properties (for example, by defining ad-hoc indexes). Moreover, the simpler functions may aim at reducing the dataset dimension by grouping objects using a rougher distance, while the more complex functions may be applied to the smaller groups found in the previous steps. In this way, the size of the input for the complex functions is reduced.

Trajectories exploration framework

The overall exploration framework we propose is outlined in the following figure.



In the extended version of the paper we shall present in details each step. Here, we limit ourselves by describing an example of exploring a set of movement data.

Clustering Example

The example dataset under analysis consists of positions of 50 trucks transporting concrete in the area of Athens, which were GPS-tracked during 41 days in August and September 2002. There are 112,300 position records consisting of the truck identifiers, dates and times, and geographical coordinates. The temporal sampling rate is regular (30 seconds.) The data are publicly available at the URL www.rtreeportal.org. From these raw data, 1100 trajectories have been reconstructed.

An analyst who needs to understand how the transportation of concrete is done in Athens has no prior knowledge about the data and about the geographic area. At the beginning, he needs an overview of the data; however, there is no appropriate visualization that would allow the analyst to see all trajectories together.

To get an idea about the origins and destinations of the trips, the analyst decides to group the trajectories by spatial closeness of their starts and ends. For this purpose, he applies a clustering tool with the distance function returning the average from the distances between the starting points and between the ending points of two trajectories. The results of the clustering show the analyst that a great majority of the trips are round trips from two places; these trips have been grouped into two clusters with 232 and 412 trajectories, respectively. Moreover, most of the trajectories in the remaining clusters either start or end in one of these places. This means that these two places play a special role as truck stations or depots. They will be henceforth referred to as depot 1 and depot 2. It is not likely that depot 1 and depot 2 belong to different companies as there are trips between them.

Next, the analyst is interested about how the trips originating from depot 1 and depot 2 are distributed over the territory and whether each depot has its “area of influence” where it dominates. The analyst applies clustering according to the positions of the trip starts (by choosing the distance function that returns the spatial distance between the starting points). In the result, the tool produces a cluster of 305 trips originating from depot 1, a cluster of 537 trips originating from depot 2, and a number of much smaller clusters. On a map display, the analyst sees that depot 1 serves mainly the northern and western parts of the territory and depot 2 mainly the southern and eastern parts, but the “areas of influence” overlap.

Now, the analyst wants to investigate the routes of the trucks from each of the depots. He would like to detect frequently occurring routes but also see how many trips were unique. The analysis is done separately for each depot. First, the analyst selects the cluster of trajectories starting at depot 1 and applies to it the clustering tool with the distance function that computes the distance according to the routes, irrespective of time. From the consideration of the resulting clusters on the map

display (Figure 1 left), the analyst gets an idea about the major routes of the trucks from depot 1. The trajectories dissimilar to all others (i.e. with the route-based distances exceeding the specified threshold) are classified as "noise". The analyst can consider these trips as well. Thus, it may be easily noted that distant trips occur less frequently than more close ones.

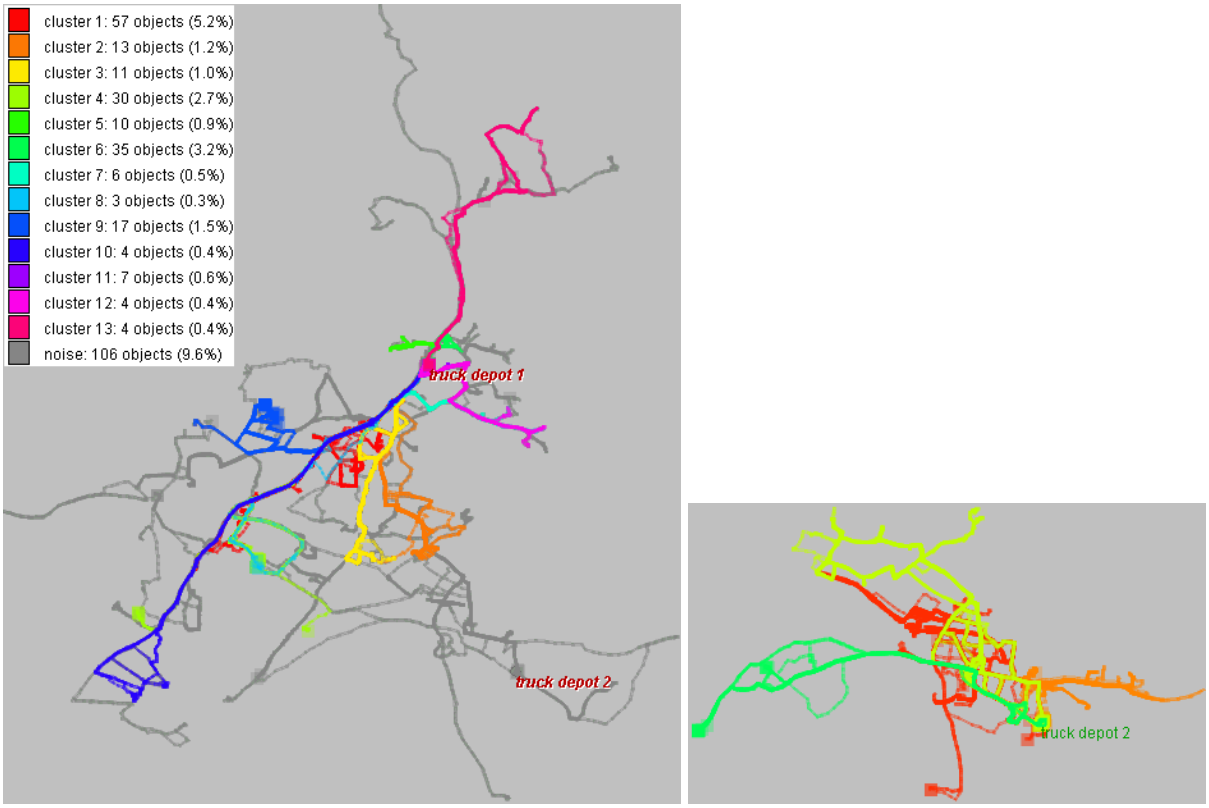


Figure 1: Results of clustering by route similarity. Left: the routes of the trucks from depot 1. Right: Four biggest clusters of trajectories from depot 2.

Analogously, the analyst examines the trips originating from depot 2. Again, distant trips occur rarely. Figure 1 right shows the biggest clusters of trajectories from depot 2 containing 166 (red), 61 (orange), 33 (yellowish), and 20 (bright green) trajectories.

For the big clusters of trajectories with similar routes, the analyst would like to see how the trajectories differ in their temporal dynamics, i.e. speeds of movement as well as times and places of stops. This analysis may be done with the help of a distance function that derives distances between trajectories on the basis of their routes taking into account the relative times when same or close locations were passed. Thus, the analyst applies clustering to the trajectories from the red cluster in Figure 2 with the use of this distance function. To understand the results, the analyst uses a 3D view where the vertical dimension represents time relative to the start of a trip. This visualization is effective for a small number of trajectories; therefore, the analyst applies it to selected clusters in order to compare them. In Figure 2, two sub-clusters are visible. In both sub-clusters, major stops occur almost in the same place on the left of the image (stops, i.e. time intervals when the position did not change, are signified by vertical lines). However, the vertical positions of the line segments between the origins (green-framed triangles on the bottom right) and the places of the stops differ for the red and cyan blue lines. The higher vertical positions of the red lines indicate slower movement and later arrival to the stop point. The red and cyan blue lines begin to diverge in the centre of the lower part of the image. By manipulating the 3D view, the analyst can find out that the loss of time mostly occurs at a crossing of two roads. The cyan blue lines avoid this crossing, which may be the reason for the better temporal performance.

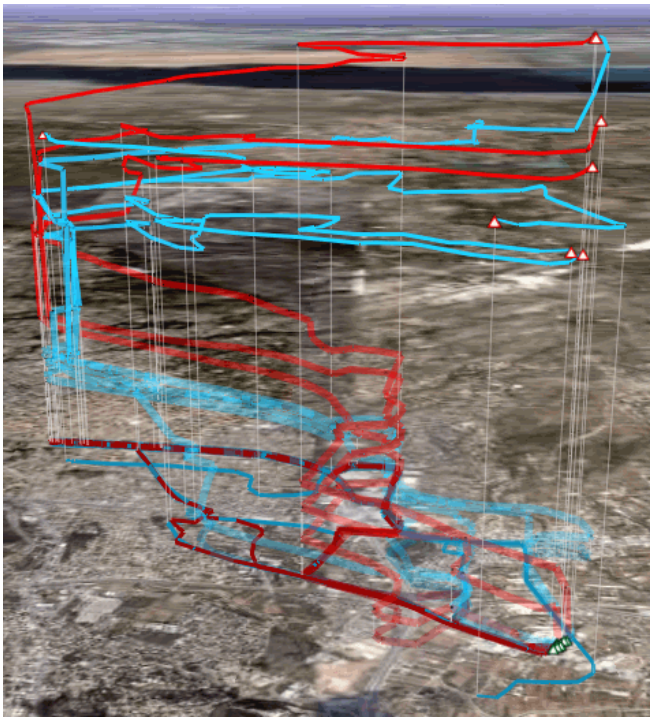


Figure 2: Two clusters of trajectories following similar routes but differing in temporal characteristics

This example demonstrates the need for diverse similarity measures in visual exploration of large collections of trajectories.

Conclusion

Trajectories of moving objects are complex spatio-temporal constructs, and analysis of multiple trajectories is a challenging task. To help a human analyst to make sense from large amounts of movement data, we suggest the procedure of progressive clustering supported by visualisation and interaction techniques. A good property of this procedure is that quite a simple distance function can be applied on each step, which leads to easily interpretable outcomes. However, successive application of several different functions enables sophisticated analyses through gradual refinement of earlier obtained results. Besides the advantages from the sense-making perspective, progressive clustering provides a convenient mechanism for user control over the work of the computational tools as the user can selectively direct the computational power to potentially interesting portions of data instead of processing all data in a uniform way. In particular, the analyst may use “expensive” (in terms of required computer resources) distance functions for relatively small potentially interesting subsets obtained by means of “cheap” functions producing quick results. In the future, we plan to extend the ideas of progressive clustering to analysis of extremely large datasets that cannot be processes in main memory of the computer. The general approach is that clustering is applied to a sample of trajectories, and then computations in the database are used to attach remaining trajectories to the clusters obtained. This approach requires efficient algorithms for checking trajectory membership in a cluster.

References

[1] Mobility, Data Mining and Privacy – Geographic Knowledge Discovery. Fosca Giannotti, Dino Pedreschi (Eds.). Springer. December 2007

[2] Gennady Andrienko, Natalia Andrienko, Stefan Wrobel. Visual Analytics Tools for Analysis of Movement Data. ACM SIGKDD Explorations, 2007, v.9 (2), pp.38-46

[3] Mirco Nanni, Dino Pedreschi: Time-focused clustering of trajectories of moving objects. J. Intell. Inf. Syst. 27(3): 267-289 (2006)
[4] Ankerst, M., Breunig, M., Kriegel, H. -P., & Sander, J. (1999). Optics: Ordering points to identify the clustering structure. In *Proceedings of ACM SIGMOD international conference on management of data (SIGMOD'99)* Philadelphia, Pennsylvania. New York: ACM.
[5] Ester, M., Kriegel, H. -P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Second international conference on knowledge discovery and data mining*, Portland, Oregon (pp. 226–231) California: AAAI.
[6] Pelekis, Nikos; Kopanakis, Ioannis; Marketos, Gerasimos; Ntoutsi, Irene; Andrienko, Gennady; Theodoridis, Yannis. Similarity Search in Trajectory Databases. 14th International Symposium on Temporal Representation and Reasoning (TIME 2007), Proceedings, IEEE Computer Society Press, 2007, pp.129-140
[7] ClusterSculptor: A Visual Analytics Tool for High-Dimensional Data, by Eun Ju Nam (Stony Brook University) and others, IEEE VAST 2007
[8] Ira Assent, Ralph Krieger, Emmanuel Mueller, Thomas Seidl "VISA: Visual Subspace Clustering Analysis" SIGKDD Explorations, December 2007, Volume 9, Issue 2