Travel Time Prediction using Machine Learning

Leone Pereira Masiero
TeCGraf - PUC–RJ
Rua Marquês de São Vicente, 225
Rio de Janeiro - Brazil
+55 21 3527-2503
leone@tecgraf.puc-rio.br

Marco Antonio Casanova
Department of Informatics - PUC–RJ
Rua Marquês de São Vicente, 225
Rio de Janeiro - Brazil
+55 21 3527-1500
casanova@inf.puc-rio.br

Marcelo Tilio M. de Carvalho
TeCGraf - PUC–RJ
Rua Marquês de São Vicente, 225
Rio de Janeiro - Brazil
+55 21 3527-2508
tilio@tecgraf.puc-rio.br

ABSTRACT
This paper investigates the application of a Machine Learning technique to predict the time that will be spent by a vehicle between any two points in an approximated area. The prediction is based on a learning process based on historical data about the movements performed by the vehicles taking into account a set of semantic variables to get estimated time accurately. The paper also describes an experiment with real-world data. Although this is preliminary work, the results were satisfactory.

Categories and Subject Descriptors
H.2.8 [**Database Applications**]: Data mining, spatial databases and GIS, and statistical databases

General Terms
Algorithms and Design.

Keywords
Support Vector Machine, Mobility Data, Travel Time Estimation.

1. INTRODUCTION
The development of new technologies, concepts and processes in the area of Information Systems and also the specific branch of Computer Science have been transformed in intensive systems in order to automate routines, methods and procedures in several areas of knowledge. The spread of these technologies has produced large volumes of data. These large volumes of data embody the knowledge of the organization, processes and environments. Access to this knowledge is the result of data analysis processes. However, database technology by itself is not comprehensive enough to support data analysis. Complementary technologies and tools are required in this process, mostly developed in the context of Data Mining and Machine Learning.

The main objective of these areas is to uncover knowledge through the analysis of data. By uncovering knowledge, we mean the discovery of patterns, trends, classes and groups of data. Recently, the areas of mobile computing, remote sensing and mobile telephony experienced fast development and, as a result, produced large amounts of data. The development of technologies for mobility management and location (GPS, for example) allows us to represent the movement of objects as one or more trajectories, consisting of an ordered set of positions over time. Trajectory data analysis is of fundamental importance to understand the behavior of objects in a given environment.

IWCTS'11, November 1, 2011, Chicago, IL, USA.

This paper aims at creating a mechanism to predict the time that a particular vehicle, associated with a driver and a region, will take to move between two distinct points. The prediction is based on a learning process based on historical data about the movements performed by the vehicles taking into account a set of semantic variables to get estimated time accurately.

This paper is organized as follows. Section 2 covers related work. Section 3 defines the methodology developed. Section 4 describes the experiments. Finally, Section 5 contains the conclusions.

2. RELATED WORK
The proposal of this paper is based on some concepts that have been studied by researchers worldwide. These concepts include semantic of moving objects trajectories, pattern detection, forecasting, location base services, and travel time prediction, among others. This section intends to present an overview of some works related to the proposal of this work.

The concept of Trajectory Pattern introduced in [1] defines a sequence of geo-referenced objects S of size m and a list of Temporal Annotations A of size (m - 1), whose values represent the temporal distance between two consecutive geo-referenced objects belonging to S. The information of the trajectory patterns, i.e., the geo-referenced objects and temporal annotations, are extracted from a set of trajectories of moving objects (in this case, a sequence of triple <latitude, longitude, timestamp>) by identifying regions of interest (geo-referenced objects) often visited by moving objects. The computation of temporal annotations is formalized as a problem of density estimate, whose values are used to calculate the time difference between timestamps of two consecutive triples in a trajectory.

Liao et al. [2] consider that a person over a period of time can follow a routine and that the routine can be "learned" by a computer system. To describe a routine, the authors consider three characteristics: location, change of transportation mode and main locations. Based on these characteristics, the authors developed a mechanism to predict the location of a person, the transportation mode changes that person will do, and the main sites and he or she will pass. The forecasting mechanism learns and infers the information using a hierarchical model to represent the state transitions (change of position, change of transportation mode and relocation, each belonging to one level of the model, which belongs to the first level lower) of a routine and particle filter-Rao Backwellised to estimate the states of the lowest level of the hierarchical model. The highest level states are based on the lowest level states. In addition, a Kalman filter is used to reduce

the error of the GPS mapping of the samples to the road network. According to the authors, this mechanism allows predictions about the indication of abnormal situations, such as traffic jams on one of the sites to be visited, or even the indication that a bus was taken by mistake.

Monreale et al. [3] authors created a mechanism called WhereNext to estimate with a certain level of accuracy the possible next locations of a moving object. The forecast mechanism uses previously extracted movement patterns, which represent possible behaviors of moving objects, such as sequences of regions often visited by a moving object from time to time. A decision tree, called a T-Tree pattern, is built on trajectory patterns (as defined in [1]) that were previously extracted and evaluated. The next indication of the location of a moving object is done by mapping the path made by the moving object until such time as a way of sub-standard T-Tree, where the children of the node of the deepest sub-mapped path indicate the possible next locations of the moving object.

Idé et al. [4] were concerned with the task of predicting travel time to an arbitrary source-destination pair on a map. The method they proposed allows the probabilistic prediction of the travel time along an unknown path (a sequence of links), such that the similarity between paths is defined as a kernel function. Their work introduces two novel ideas. The first one is the use a kernel string to represent the similarity between paths. The second one is the application of the Gaussian process regression for predicting the travel time.

Wu et al. [5] have similar goals, but their approach is different. The authors use Support Vector Regression (SVR) to estimate the time that a particular stretch of highway will be covered, based on data collected by speed gauges posted along specific roadways in Taiwan. Also according to the authors, SVR had better results than Artificial Neural Networks because SVR is more amenable to generalization than Artificial Neural Networks.

The work of Spaccapietra et al. [8] is used as a theoretical base to be extended to define the types and patterns of trajectories. The paper presents a semantic definition to paths based on stops and moves, where the stops represent the important parts for the semantics and also analyses the needs from the point of view of the application in terms of data modeling for adding semantic value to the moving objects trajectories. This is accomplished through two approaches to conceptual modeling trajectories being perceived as elements of the first order: a proposal by a design pattern and the other in the definition of owner types suitable for trajectories, where both can be extended with semantic information in the context of application. Both approaches are intended to provide rules for builders and designers of databases that use mobility data. These modeling seek to provide a characterization to trajectories and its components with attributes, semantic constraints, topological constraints and links to application objects. The two contributions of modeling are demonstrated through the presentation of each schemes adapted to the context of bird migration.

The works [9] and [10] of Alvarez et al. use the definition of the conceptual model of trajectories with segmentation by stops and moves proposed in the work cited above [8]. The first makes the discovery of movement patterns for trajectories based on data mining techniques. It proposed a framework to model and perform the mining for discovery of movement semantic patterns. The main focus is on discovering the most frequent moves between two stops, where each stop is seen as an application's interest, which provides better understanding of the behavior of movements in geographical areas and facilitates writing queries about the patterns of moving objects. The second, taking into account the complexity of queries and analysis of trajectories data in raw format, seek to add semantic information to the trajectories based on the geographical context in which they operate. For this, it is proposed a generic model to represent the parts of trajectories that are important to the application as well as an algorithm to compute these parts, which in the model are stops transformed into geographic objects that have meaning for the application. Analyses are then performed on these data, minimizing the spatial search and spatial joints, therefore, can obtain a significant reduction in the complexity of queries for semantic analysis of trajectories.

Still based on the conceptual model of Spaccapietra et al. [8], the works of Palma et al. [11] and Rocha et al. [12] followed by clustering approaches to knowledge discovery. The first proposes a solution to the discovery of important places in the trajectory based on speed, i. e., the stops discovery. The work is based on the simple idea that segments with lower speed may represent local interest in two parts: the first part of the process operates in the discovery of potential stops and the second based on the outcome of the first, analyzes them related the geographical information. The second seeks to discover places of interest based on changes of direction, considering areas where this aspect is important as the discovery of places where the ships perform sea fishing, preventing them engaging activity in forbidden places.

On another line of work, Guc et al. [13] supports the idea that the trajectory data can be used to facilitate the manual process of trajectories semantic annotation. For this, they propose a trajectory annotation model based on notion of episodes that allows the separation between the physical and semantic part and also an architecture to program to perform semantic annotations. Yan et al. [14] work stays on the same line and proposes a framework (SeMiTri) of general propose to various domain applications (i.e. to heterogeneous trajectories) that lets you manage and enrich trajectories with semantic annotations, allowing the application can benefit from a semantic representation of movement through the inferences made from space-time properties of the position raw data (e.g. extraction of stops and moves, tracking its direction or movement pattern), geographical regions covered by the trajectories (e.g. streets and notables local) and application objects related to the trajectories (e.g. customers, parking). The framework takes advantage of the proposed model for semantic trajectories, where a trajectory is represented as a sequence of semantic episodes that correspond to a interpretation of the application and also presents three algorithms for performing trajectory annotations, one based on regions of interest, another based on the path and the latter based on points of interest, these algorithms are responsible for covering the peculiarities of the heterogeneity of trajectories, as trajectories of vehicles, people walking, animals, etc.

3. METHODOLOGY

The methodology we propose in this paper uses a mechanism for estimating the time that will be spent by a vehicle between any two points that: (1) pre-processes raw trajectory data to create aggregated data; and (2) learns the behavior of the vehicles using this aggregated data.

3.1 Moves and stops

Any moving object moves and stops. The stops have a goal (semantic) and a period of time: a person stops for 10 minutes at a

bus station to take a bus or a vehicle stops for 5 minutes in a gas station to refuel. Figure 1 illustrates a situation in which a moving object made three stops P1, P2 and P3 along the path between A and B in the space S.

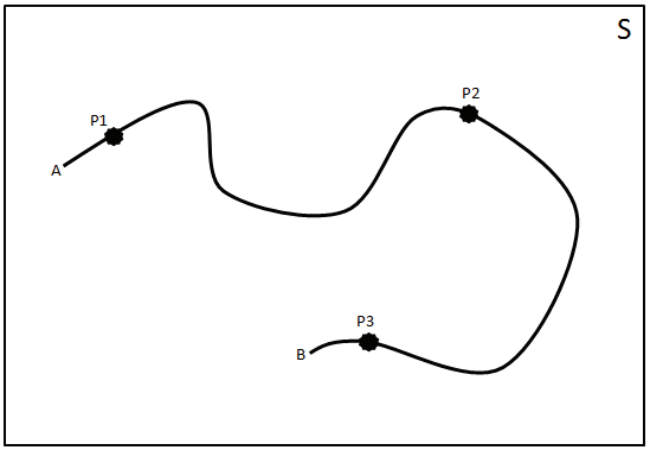

Figure 1 - Path example with stops (P1, P2 e P3).

Taking into account the movements and stops (later on the concept of stop is defined) of moving objects, the mechanism of travel time prediction proposed in this paper uses data mobility (GPS coordinates with timestamp and speed) to "learn" the behavior of vehicles (and their drivers) in a given region and estimate the time a vehicle (driven by the same driver) in a given region will take to move between two distinct points.

The construction of the travel time estimating mechanism involves the following steps: extracting the mobility data; processing the trafficked segments; and creating the time estimation model. These steps are described in the next subsections.

3.2 Mobility data extraction

The mechanism of travel time estimation is based on the historical movements of vehicles over time, as the goal is to estimate the travel time of a vehicle, associated with a driver, between two points. Thus, it is necessary to know when and where the vehicle has moved and when and where it has stopped. This is done by pre-processing the data to create more qualitative information indicating moves and stops (figure 2) when compared to simply positioning the vehicles in time (figure 3). A stop is detected when the vehicle does not travels *s* meters in *t* minutes, where *s* and *t* are parameters of the stop detection function and the stop location is a known place based on the application context

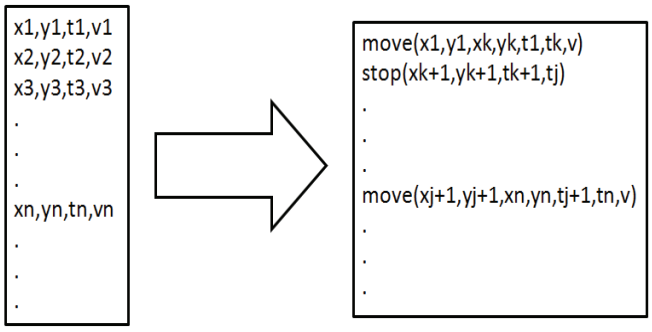

Figure 2 - Extraction of data movement through the positioning data.

x	y	t	v
-46,7864	-23,1287	2010-08-11 12:33:54	45,4
-46,7790	-23,1265	2010-08-11 12:34:54	23,8
-46,7835	-23,1276	2010-08-11 12:35:54	0,0
-46,7835	-23,1276	2010-08-11 12:36:54	0,0
-46,7835	-23,1276	2010-08-11 12:37:54	0,0
-46,7835	-23,1276	2010-08-11 12:38:54	0,0
-46,7835	-23,1276	2010-08-11 12:39:54	0,0

Figure 3 - Sequence of a moving object positions.

The model illustrated in figure 5 shows the relationship between *move*, *stop*, *vehicle*, *driver* and *region*. Whereas the data are divided by regions, every movement and stop of a vehicle are associated with a region and a driver.

3.3 Trafficked segment processing

On the path between two stops of a vehicle there can be a variation of vehicle speed, causing some parts to be slower than others. Segmenting portions of the movement of vehicles allows us to represent this difference in speed over a movement. This segmentation allows to estimate accurately know how long time the vehicle will spend from the current position to the next known place (again, based on the application context) location it will stop considering slices of the segment with different average speed. The segmentation is done in sections, as illustrated in figure 5. Considering the movement between stops P1 and P2, four segments are defined: S1, S2, S3 and S4. Segment S1 starts at the stop P1 and ends at the stop P2. Segment S2 starts at the position indicated in the figure and ends at stop P2. Segment S3 starts at the position indicated in the figure and ends at stop P2. And finally, segment S4 starts at the position indicated in the figure and ends at stop P2. Another way to segment the movement would be as follows: the first segment would begin at P1 and end at S2. The second segment would begin at S2 and end at S3. The third segment would begin at S3 and end at S4. And finally the last segment would begin at S4 and end at P2. But the first targeting option was chosen.

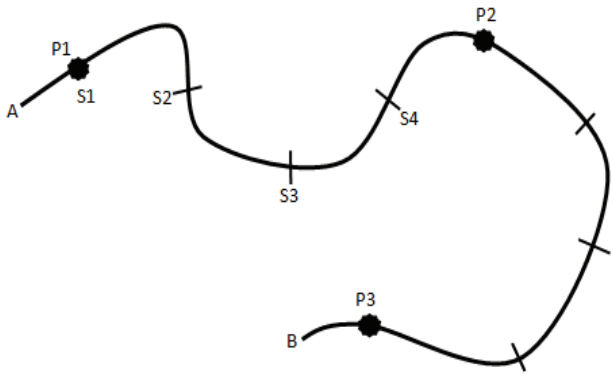

Figure 4 - Illustration of movement segmentation.

The segmentation of movements creates a new entity in the model illustrated in figure 5: the *Segment* entity. Thus, a movement (*Move*) has a set of entities *Segment*. The entity *Segment* has as attribute the *start point* (or coordinate), the *end point* (or coordinate), the *start timestamp*, the *end timestamp* and the *distance* traveled.

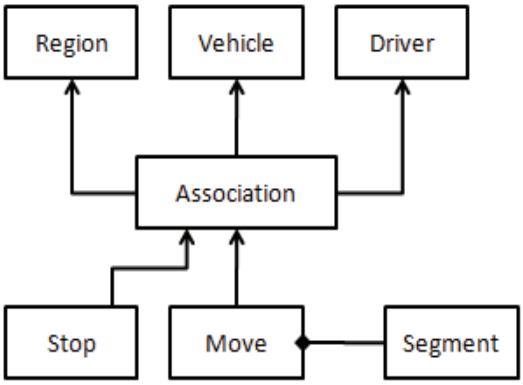

Figure 5 - Diagram of relationship between data model classes.

3.4 Creating the travel time estimation model

The time estimation model is a Machine Learning model that takes into account some variables associated with the vehicle's movement. Since the goal of the model is to estimate the time a vehicle, driven by a driver in a given region, will spend to travel a path between two points, the model variables are: the latitude and longitude of the segment start point, the latitude and longitude of the segment end point, the distance to be traveled between these two points (segment distance), the segment start timestamp's hour of day, the segment start timestamp's date, the segment start timestamp's day of the week and the identifier of the vehicle. The use of these variables is the key of this work, because it is important to know when and where the vehicle is. This information could indicate accurately how long time the vehicle will spend to travel the current segment. For example, if the current time is a rush, the vehicle may spend more time to travel the current segment than in other normal time in the same place. The same segment could have different estimated times depending on the driver, vehicle and current time. A driver can be faster than others, or a vehicle can be older and slower than others. For each region, a model is created with these variables. To create the time estimation model, we used SVR (Support Vector Regression) [5,6]. SVR is based on the concept of supervised learning, for which a given set of input data, where each entry has the value to be estimated and the values of the variables of the model, "learns" the behavior of the data through the training and testing process. Very briefly, SVR defines a weight for each variable. The training process changes the values of these weights so that the value estimated by the model approaches the estimated value of the entry, and the testing process shows how the model "learns" the behavior of the data that were used in training process. According to [5], for this application, SVR has better results than Artificial Neural Networks because SVR is more amenable to generalization than Artificial Neural Networks. In [6] SVR is presented in more details.

The SVR input data are extracted from the model shown in figure 5. Each segment (Segment entity) represents an instance of the set of input data of SVR. Considering the input data as a table, each table row is represented by a segment where a column of the table is the value to be estimated, in this case, time, and the other columns are the values of the model variables, as illustrated in figure 6. Each variable value is extracted from the attributes of the segment: the time taken to traverse the path between the start and the end points (t); the latitude (y1) and longitude (x1) are extracted from the start point; the latitude (y2) and longitude (x2) are extracted from the end point; the distance (d) between these two points is calculated using the Bing Maps Routes API [7]; the hour of the day (h), the day of the month (dm) and the day of the week (dw) are taken from the start timestamp of the segment. To estimate the time using the proposed estimation model, the following values must be input: start point, end point, hour of the start point timestamp, day of week of the start point timestamp, day of the month of the start point timestamp and the vehicle's identifier. The model will return the estimated time that the vehicle will take to travel between the start and end points taking into account the start point timestamp.

t	x1	y1	x2	y2	d	h	dw	dm	v	dr
t(S1,P2)	x(S1)	y(S1)	x(P2)	y(P2)	d(S1,P2)	h(S1)	dw(S1)	dm(S1)	v1	d1
t(S2,P2)	x(S2)	y(S2)	x(P2)	y(P2)	d(S2,P2)	h(S2)	dw(S2)	dm(S2)	v1	d1
t(S3,P2)	x(S3)	y(S3)	x(P2)	y(P2)	d(S3,P2)	h(S3)	dw(S3)	dm(S3)	v1	d1
t(S4,P2)	x(S4)	y(S4)	x(P2)	y(P2)	d(S4,P2)	h(S4)	dw(S4)	dm(S4)	v1	d1
...										

t	x1	y1	x2	y2	d	h	dw	dm	v
t(S1,P2)	x(S1)	y(S1)	x(P2)	y(P2)	d(S1,P2)	h(S1)	dw(S1)	dm(S1)	v1
t(S2,P2)	x(S2)	y(S2)	x(P2)	y(P2)	d(S2,P2)	h(S2)	dw(S2)	dm(S2)	v1
t(S3,P2)	x(S3)	y(S3)	x(P2)	y(P2)	d(S3,P2)	h(S3)	dw(S3)	dm(S3)	v1
t(S4,P2)	x(S4)	y(S4)	x(P2)	y(P2)	d(S4,P2)	h(S4)	dw(S4)	dm(S4)	v1
...									

Figure 6 - Example of input used in the training algorithm associated with the first segment of the figure Figure 4.

4. EXPERIMENTS

At the Computer Graphics Laboratory of the Department of Informatics of the Catholic University of Rio de Janeiro (TeCGraf / PUC-RJ), we are developing a vehicle tracking project supported by a subsidiary of PETROBRAS, a Brazilian petroleum company. The vehicles have a tracking device that sends position data to a central server every 30 seconds.

The tracked vehicles transport liquid gas. Each vehicle is based on a distribution base, which serves as a particular region. The vehicles leave the base to deliver the product to the company's customers and, at the end of the service, they return to the base (figure 7). The departure and the return of the vehicle to the base are defined as a trip. For each trip, there is a delivery schedule, which the vehicles must be met. Therefore, when leaving the base, each driver knows which customers he must attend, and when.

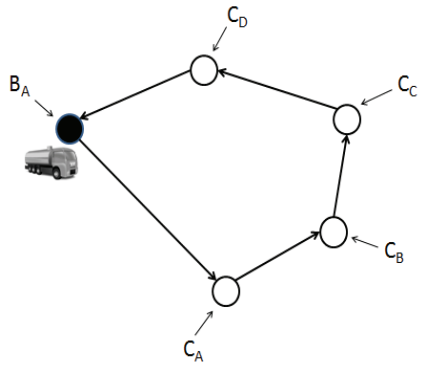

Figure 7 - Illustration of a trip.

Applying the data from this project to the proposed model, as shown in figure 5, the moves are represented by movements

between the base and client, between the customers and finally between the client and the base, and the stops are represented by the stops at the base and at customers.

Performing the steps proposed in our methodology, we built a database of mobility of these vehicles, which served as input to the process of training and testing the time estimation model. To train and test the model, it was used the cross-validation method which uses different parts of the input to train and test the model. The input is divided in n parts where (n – 1) parts are used to train and the last one is used to test. The training and testing processes are executed n times where each testing execution uses a different input. The model error is calculated in each training and testing execution. The testing execution is done by calculating the difference between each estimated time from the trained model and input target time. The error is the mean square error.

The provided vehicle data for the experiments belong to a single base. There were 1,800 trips and a little more than 15,000 stops during a period of 15 months. The model final error was 6%. That is, when estimating the travel time between two points, the model was wrong for about 3.6 minutes. These results are preliminary since we may still use other variables to give the model a better learning capacity. The same data were used to calculate the average estimated time, whose mean square error was 49%. In this case, the average was calculated for all entries in the database without considering the values of the variables. Obviously one can calculate the average values for each combination of variables from the model, but in this case the more variables the model has the larger the number of combinations of variable values will have to be used to compute the average.

5. CONCLUSIONS

The travel time estimation for moving vehicles involves a large number of variables, which makes the solution complex. Changes in traffic behavior greatly influence the travel estimated time of the vehicle. But the vehicle driver itself, the vehicle's features, the vehicle load, if the day is before a holiday, are also variables to be considered, in addition to the vehicle identifier.

Another point to be discussed is the availability of data to perform such experiments. The experiments described in this paper were made possible by data kindly provided by the vehicle monitoring project developed for Petrobras.

Finally, the methodology developed in this paper has not yet been fully tested with real-time data, but preliminary results proved to be reasonably satisfactory. In addition, we intend to use other techniques such as Artificial Neural Networks, regression and Principal Component Analysis to estimate travel time.

6. REFERENCES

[1] Giannoti, F., Nanni, M., Pinelli, F. and Pedreschi, D. "Trajectory pattern mining". KDD 2007, p. 330-339.

[2] Liao, L., Patterson, D., Fox, F. and Kautz, H. "Learning and inferring transportation routines". Artificial Intelligence, v.171 n.5-6, p.311-331, Abril, 2007.

[3] Monreale, A. et al. "WhereNext: a location predictor on trajectory pattern mining". KDD 2009, p. 637–646, 2009.

[4] Idé, T. and Kato, S. "Travel-Time prediction using Gaussian process regression: a trajectory-based approach". SIAM Intl. Conf. Data Mining (2009).

[5] Wu, C-H., Wei, C-C., Ming-Hua Chang, M-H., Su, D-C. and Ho, J-M. "Travel Time Prediction with Support Vector Regression". Proc. Of IEEE Intelligent Transportation Conference. October, 2003 pg. 1438-1442.

[6] http://en.wikipedia.org/wiki/Support_vector_machine, "Support Vector Machine", accessed on 08/14/2011.

[7] http://msdn.microsoft.com/en-us/library/ff701705.aspx, "Routes API", accessed on 09/14/2011.

[8] Spaccapietra, S., Parent, C., Damiani, M. L., de Macêdo, J. A., Porto, F., and Vangenot, C. A conceptual view on trajectories. Data & Knowledge Engineering 65, 1 (2008), 126-146. Including Special Section: Privacy Aspects of Data Mining Workshop (2006) - Five invited and extended papers.

[9] Alvares, L. O., Bogorny, V., de Macêdo, J. A. F., Moelans, B., and Spaccapietra, S. Dynamic modeling of trajectory patterns using data mining and reverse engineering. In Tutorials, posters, panels and industrial contributions at the 26th International Conference on Conceptual Modeling - ER (Auckland, New Zealand, 2007), A. H. F. L. L. M. John Grundy, Sven Hartmann and J. F. Roddick, Eds., vol. 83 of CRPIT, ACS, pp. 149-154.

[10] Alvares, L. O., Bogorny, V., Kuijpers, B., de Macêdo, J. A. F., Moelans, B., and Vaisman, A. A. A model for enriching trajectories with semantic geographical information. In Proceedings of the 15th ACM International Symposium on Geographic Information Systems (New York, NY, USA, 2007), GIS '07, ACM, pp. 22:1-22:8.

[11] Palma, A. T., Bogorny, V., Kuijpers, B., and Alvares, L. O. A clustering based approach for discovering interesting places in trajectories. In Proceedings of the 2008 ACM symposium on Applied computing (New York, NY, USA, 2008), SAC '08, ACM, pp. 863-868.

[12] Rocha, J., Oliveira, G., Alvares, L., Bogorny, V., and Times, V. Dbsmot: A direction-based spatio-temporal clustering method. In Intelligent Systems (IS), 2010 5th IEEE International Conference (july 2010), pp. 114-119.

[13] Guc, B., May, M., Saygin, Y., and Korner, C. Semantic annotation of gps trajectories. 11th AGILE International Conference on Geographic Information Science (AGILE), 2008. (2008).

[14] Yan, Z., Chakraborty, D., Parent, C., Spaccapietra, S., and Aberer, K. Semitri: a framework for semantic annotation of heterogeneous trajectories. In Proceedings of the 14th International Conference on Extending Database Technology - EDBT (New York, NY, USA, 2011), EDBT/ICDT '11, ACM, pp. 259-270.