Mining Traffic Patterns from Public Transportation GPS Data

Florin Lipan and Adrian Groza
Technical University of Cluj-Napoca
Department of Computer Science
Baritiu 28, RO-400391 Cluj-Napoca, Romania
Email: florin.lipan@student.utcluj.ro, adrian.groza@cs.utcluj.ro

***Abstract*—Bus schedules submitted by public transportation companies often lack accuracy. Generally this relates to traffic being inconsistent over most parts of the road network. We set on finding a model for predicting these speed variations over GPS monitored bus routes, by using association rules, and then apply it on the local bus network of Cluj-Napoca, Romania.**

***Index Terms*—public transportation; GPS fleet monitoring; association rules; traffic patterns;**

I. Problem Statement

One aspect of public transportation that makes up a good premises for data mining is *regularity*: buses run in cycles (laps) along the same track, attending every station according to a fixed schedule or at a given frequency [1]. Therefore, with every performed cycle, this feature will enable us to collect a large amount of information on terrain layout or traffic.

If we were to take into account only road features (length, speed restrictions, terrain geometry etc.) and the daily cycle of buses, predicting the schedule of a public transportation system should be possible. But often the variations in traffic make it very difficult to determine an acceptable approximation to the arrival and departure times. Rush hours, weekends or holidays are common exceptions to the traffic flow that may be a source of unpredictable behaviour. Moreover, some features along a bus route - like traffic lights and pedestrian crossings - represent for the driver elements of randomness which can be foreseen only with a certain probability. Hence a reliable integrated prediction system is needed, one that should take into account both the relationship between traffic events and date & time information, as well as the probability of random events like stopping at the traffic lights.

The current paper sets on finding a model for speed variations of buses, using data mining techniques on collected GPS information from public transportation. The goal of this research is to investigate correlations between road features, date & time information, passenger count etc. and speed.

II. Technical Instrumentation

The *NMEA format* is a standard interface developed by the National Marine Electronics Association (in the U.S.) for communications between marine electronic equipment, including GPS receivers. GPS GGA (Global Positioning System Fix Data) sentences contain the following information: i) UTC timestamp in "hhmmss.sss" format (hh - hours, mm - minutes, ss - seconds, sss - miliseconds); ii)latitude and longitude in "Dm,H" format (D - degrees, m - minutes with 4 decimals precision, H - hemisphere); iii) GPS quality information: 0 for "invalid", 1 for "GPS fix", or 2 for "DGPS fix"; iv) number of satellites being tracked; v) horizontal dilution of the position: a measure of GPS accuracy, based on the geometry of tracked satellites; this attribute assumes a numerical value between 1 and 20, where "1" equals to the highest possible confidence level; vi) altitude, followed by its unit of measure.

The *Haversine formula* is used for calculating geographical distances, or more explicitly - to compute the shortest path between two geographical coordinates on the surface of the globe (measured in degrees or radians). Let (ϕ_s, λ_s) and (ϕ_f, λ_f) be the (latitude, longitude) pair of two points s and f, measured in radians. Then the spherical (angular) distance of the two will be:

$$\Delta\hat{\sigma} = 2arcsin(\sqrt{sin^2(\frac{\Delta\phi}{2}) + cos\phi_s cos\phi_f sin^2(\frac{\Delta\lambda}{2})}) \quad (1)$$

where $\Delta\phi = \phi_f - \phi_s$ and $\Delta\lambda = \lambda_f - \lambda_s$. The distance d (or arc length), for a sphere of radius r and $\Delta\hat{\sigma}$ becomes $d = r\Delta\hat{\sigma}$.

III. Data Acquisition

Initially, GPS devices are mounted on one or several vehicles along the examined bus route. The movement of a vehicle is going to be sampled by the GPS receiver at a given time rate, thus associating every piece of geographical information (latitude, longitude, altitude) with a timestamp. The devices should also offer the possibility to communicate with a remote server, task usually accomplished by GPRS, 3G or radio connections. As the GPS device records geographical information, it is sent via the remote connection to a central processing unit, which accomplishes storing, prepossessing and analyzing the data. Also by the means of this connection, the GPS device can be used not only for data mining but also for real time monitoring of the bus fleet.

How long and how often we keep monitoring the activity on a certain bus line is a question of what kind of results we are actually expecting. Traffic generally varies from workdays to weekends, and also according to the time of the day (e.g. rush hours) and to seasons (e.g. snow-days in northern Europe or Monsoon season in India). For the latter case, our analysis will have to cover the whole year on a regular basis: monitoring

978-1-4244-8230-6/10/$26.00 ©2010 IEEE

Fig. 1. Cluj-Napoca - GPS log over the first six bus stops of line “35”.

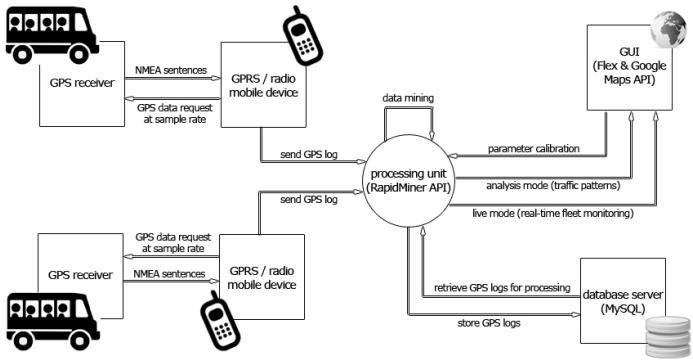

Fig. 2. Top view system architecture.

a couple of days per week, for the whole year. Both for limited studies as well as in the case of year long surveys, the following rules should be considered: i) the analysis should preferably cover the entire daily schedule on the given route; otherwise omitted time periods will be assigned to nearest available data and results for these gaps might be unreliable; ii) there should be a balance between the number of samples taken on workdays and weekends; iii) whether we provide GPS coverage for only one bus on the track or for all of them, at the end of the survey there should be enough data available to minimize the effect of outliers and technical errors; *horizontal dilution* can be used to filter unreliable information.

Our experiment focused on the Nokia LD-3W, a low-cost Bluetooth GPS device, which can be paired up with any Bluetooth & GPRS enabled, Java MIDP 2.1 mobile phone. We used the popular Nokia 2700 phone. For establishing a communication line between the LD-3W module and the mobile device, we developed a Java midlet. The application accomplishes four main tasks: 1) creates and manages a Bluetooth connection between the mobile phone and any GPS device within signal range; 2) receives NMEA strings from the GPS device at a given sampling rate (between 0.5 seconds and 2 minutes), parses this information (latitude, longitude, altitude, timestamp, signal quality, horizontal dilution, number of available satellites) and appends it to a log file; 3) whenever running in “LOG MODE”: at the end of the transmission, the midlet converts the log to KML and stores it on the phone’s memory; the data is exploited later by the processing unit, for data mining purposes; 4) whenever running in “LIVE MODE”: every parsed piece of information from the GPS module is sent in real time to a remote server (the *processing unit*) via the GPRS Internet connection of the mobile device.

IV. System Architecture

Fig. 2 depicts the architecture of our system. On the left side we have the *data acquisition system*, described on the previous section. The *processing unit* coordinates all data mining activities and is responsible for drawing the actual conclusions (rules) out of the knowledge base. It implements RapidMiner methods through the RapidMiner API.

TABLE I
Raw GPS Information.

ID	Date & Time	Latitude	Longitude	Altitude	Dilution
1	2010-03-18 09:10:22	46.755738	23.593575	431.2	2.0
2	2010-03-18 09:10:27	46.755780	23.593848	437.7	2.0
3	2010-03-18 09:10:32	46.755833	23.594278	445.6	2.0

A. Feature Selection and Combination

Data acquisition offered us valuable knowledge on the geographical coordinates (*latitude*, *longitude*, *altitude*) of the monitored vehicles, associating them to *date & time* references. We can also assess data reliability using the *horizontal dilution* attribute. Table I shows a sample of the initial state of our knowledge base, as acquired from KML logs.

The finality of this research should be finding correlations (rules) between road features, time information, bus load etc. and the speed variations that actually stop us from assuming a fixed bus schedule. We shall combine the current knowledge in order to assess new features that would better explain the speed variations. By applying the Haversine formula on every two consecutive points we can obtain *distance*. Correlating this feature with the timestamp reference of consecutive points leads us to *speed*, which is a determinant attribute to our system. Data information can be processed to obtain the *day of the week* or the *season of the year*. *Altitude difference* of two consecutive points could also be of help, as speed usually increases over the descent and drops when driving uphill.

Another interesting feature that can be obtained from the raw GPS data sets is *bus station interest*. This feature automatically assesses passenger flow at bus stops and is deduced mainly from the time spent by buses loading and unloading passengers in every station. The more time a vehicle spends waiting at a particular station, the higher the passenger flow (*in* and *out*) and subsequently the higher the interest for this station, at the given timestamp. Just think about the crowding in downtown bus stops at rush hours compared to the flow in suburban stations on weekends. In order to automatically label a geographical point as being part of a *bus stop* we

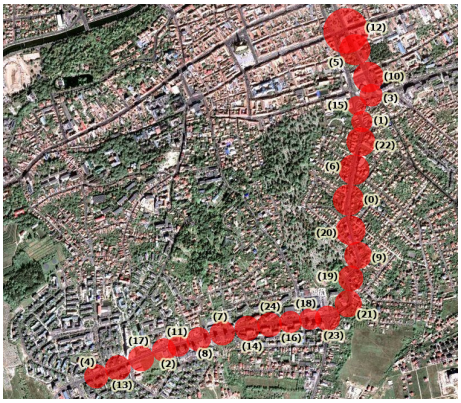

Fig. 3. Labeled clusters resulted from road partioning via k-Means.

must assume a consistent volume of GPS information. For every log we process, speed is always going to drop to 0 around bus stations. Of course, this is also the case of road sections with traffic lights, pedestrian crossings, traffic jams etc. However, given a consistent knowledge base, we are able to make the following observation: whilst speed *always* levels to 0 around bus stops (given all stations are mandatory for the bus driver), for all other situations this will happen only by a certain probability (e.g. traffic lights and pedestrian crossings will generally have random effects, traffic jams only take place at rush hours). Thus, points which maintain a constant speed of 0 over all GPS logs will be labelled as bus stations and we may calculate their *interest* feature.

B. Feature Discretization

Initially, coordinates have been converted to decimal degrees with a 6 decimals precision (every 0.000001 decimal degree provides a 0.111 meters accuracy). We would like to maintain this precision on every new feature that derives from geographical coordinates, like for example *speed*. Because some data mining algorithms (e.g. FP-Growth) can not process continuous datasets, the features need to be discretized.

First we attend the set of geographical coordinates: we want to partition the whole length of the monitored bus route into smaller clusters. Each cluster should preferably cover up a road section of 25 - 50 meters, as we would like to approximate speed to this interval rather than to every couple of points. The size of the bins will have to vary according to the number of available example sets (use smaller bins - with higher precision, for a large number of examples or use wider bins for less data and consequently less accurate conclusions). The clustering operation can be accomplished with k-Means, where *k* will be a function of *road length* and the *number of available example sets*. Fig. 3 depicts the outcome of clustering the geographical coordinates space using k-Means over 25 bins.

Other features will be discretized by frequency: *speed*, *altitude difference*, *time of the day*. This operation takes as an input the continuous set of data and the number of bins to create, which is a function of the *number of examples*, the *feature type* and the *expected result accuracy*. Areas with high information density will contain narrower, more accurate bins, whilst sections with less data available will have wider bins. If there are large amounts of data available to us, we can afford a high number of bins; otherwise precision will have to suffer.

TABLE II
DISCRETIZED DERIVED FEATURES.

ID	WD/WE	Minute / Day	Speed	Road Cluster
1	wd	range3 [543 - 576]	range5 [7.35 - 12.46]	16
2	wd	range3 [543 - 576]	range6 [12.46 - 18.39]	16
3	wd	range3 [543 - 576]	range7 [18.39 - 24.12]	18

Attributes like *day of the week* and *season of the year* can be discretized by specification. For example, days can be divided into "workdays" and "weekends". A sample of such discretized features is depicted in Table II, where wd stands for work day. Note that this information is deduced from the raw data presented in Table I. Road clusters relate to fig. 3.

C. Finding Association Rules

We have selected and discretized the features that best relate to the speed variations on our track. Now we want to be able to draw some conclusions out of this information. We are going to use *association rules*. The FP-Growth algorithm will help us identify frequent itemsets. *Minimum support* value for FP-Growth needs to be adjusted according to the number of available example sets, but also to the size of the bins generated through discretization and to the data reliability degree we are aiming for. We suggest using the following formula in order to estimate the minimum support:

$$minsup = \prod_{f=1}^{N} \frac{\alpha_f}{|bins_f|} \tag{2}$$

where N is the number of selected features and $|bins_f|$ is the number of bins (partitions) created for feature f. Variable α_f depicts the degree at which examples are equally distributed within the bins of feature f. It takes values between 1 (the highest degree of equal distribution) and $|bins_f|$ (the lowest degree). If we strive to balance the number of examples among all time intervals and all road clusters, α_{time} and α_{road} should be closer to 1. However, there are other features to which this kind of approach would be nonsense. For example, it's highly impossible that we are going to get an equal distribution of all ranges of *speeds* over the same road section. Variable α_{speed} will probably be closer to $|bins_{speed}|$ in this case.

After having generated frequent itemsets, they are fed to the association rules operator, which takes the *minimum rule confidence* as an argument. The result of this operation is a set of rules based on frequent itemsets. Because our initial goal was to infer the causes of speed variations, we are only interested in rules that take *speed* as a conclusion.

V. RESULTS

Fig. 1 shows the first six bus stops of bus line "35", based on satellite imagery of Cluj-Napoca, Romania. 42 GPS logs have been acquired on this track, corresponding to 42 different

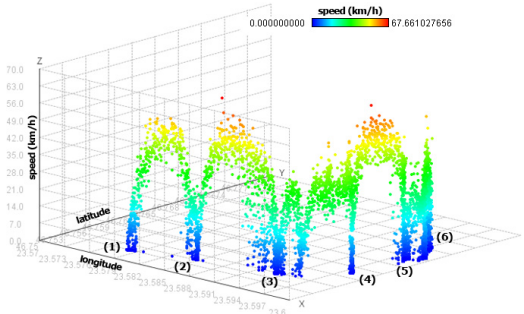

Fig. 4. Knowledge representation for 42 GPS logs.

TABLE III
ASSOCIATION RULES INFERRED OVER LINE "35".

No.	Premises	Conclusion	Support	Confidence
1	road_cluster = 10	speed = [0-0]	0.047	0.351
2	road_cluster = 1, day = wd, time = [9:06-9:49]	speed = [36-41]	0.001	0.238
3	road_cluster = 1, day = wd, time = [9:49-13:04]	speed = [41-47]	0.001	0.352
4	road_cluster = 6, day = wd	speed = [47-68]	0.014	0.779
5	road_cluster = 23, day = wd	speed = [13-18]	0.016	0.338
6	road_cluster = 16, time = [8:44-9:06]	speed = [0-0]	0.004	0.221

bus rides. All 4747 points contained within these logs have been plotted on fig. 4; bus stops are labeled using numbers and speed is represented over the Z-axis as well as in colours. This graph confirms our observations on speed variations. One of the rides has also been depicted with fig. 1: every coloured point represents one piece of GPS information, sampled from the GPS receiver at the frequency of 5 seconds. Because the time interval between two consecutive points is a constant value, it is possible to assess speed only from point densities. However, in order to make this interpretation easier we've also added a reference to speed, which is shown in colours.

The current knowledge base is not consistent enough to make complex assumptions over traffic patterns. Therefore, our analysis is going to use only a limited number of attributes: *road partition* (road cluster), *speed*, *time of the day* and *day of the week*. Feature discretization will produce 25 road clusters, 12 speed clusters and 5 time intervals. By applying Formula 2 we round up the minimum support to the value of 0.001; confidence will be leveled to 0.2.

Table III shows a sample of rules identified by our system. Only rules that take speed as a conclusion have been retained. Road clusters correspond to labels on fig. 3. Rule no. 1 takes the most general form: only the road segment in the premises and speed in the conclusion. It has successfully identified bus station *(5)*, where speed can be approximated to 0. Rule no. 6 states that speed will drop to 13-18 km/h within cluster 23, around workdays; this is valid because the road partition corresponds to a major intersection, where traffic jams often occur. However, the most important form of inferred rules are the ones pointed out with rule 2 and 3: both rules correspond to the *same road cluster* but referr to different time periods and different conclusions - these are the *speed variations* we have been looking for. Both point to cluster 1 which is right next to clusters 15 and 3, important sources of traffic disruption around rush hours. Finally, rule no. 4 states that high speeds can be attained with cluster 6; this is also expected, because the segment corresponds to a straight four lane descent.

VI. Discussion and Related Work

Provided that a sufficient volume of information exists, the presented system should be able to assess a set of reliable rules for predicting traffic behaviour. Until now we were interested in associations that led to speed variations, but other valuable rules might also be inferred out of the knowledge base. For example we could determine the relationship between date & time values and passenger flow (or *bus station interest*).

Potential benefits include: 1) achieve a better coordination of the bus fleet: by knowing the correlations between the time of the day and the speed over each road section, buses running on the same route won't overlap their schedules any more; 2) offer a reliable bus schedule to both passengers and crew: the inferred rules could help put up a dynamic schedule on the company's website; 3) improve quality of service by determining high interest bus stations, aiming better coverage at critical hours; 4) balance fuel consumption and enable better duty scheduling and duty rostering [2].

[3] uses smart-card information to determine the *variability of public transit use*, by the means of data mining techniques. Results describe the correlations between fare category and day of usage or boarding hours and frequently used bus stops. With [4], the authors suggest a GPS data management system for GPS monitored buses, by deriving travel time patterns from historical data, and apply these patterns on real-time situations, with the corresponding adjustments. The authors focus on *time* values, whilst our system is based on time *derived* features, like *speed*, providing more knowledge than raw GPS data.

VII. Conclusion

Variations in traffic might prevent us from assuming a fixed schedule over a certain bus line, but the current paper has demonstrated that traffic generally follows patterns and that these patterns can lead to accurate predictions of bus arrival times. On going work regards the combination of the above results with the activity theory to identify solutions for encouraging people to use public transportation [5].

Acknowledgment

This work was supported by CNCSIS-UEFICSU, project number PNII-Idei 170/2009.

References

[1] A. Schöbel, H. W. Hamacher, A. Liebers, and D. Wagner, "The continuous stop location problem in public transportation networks," *APJOR*, vol. 26, no. 1, pp. 13–30, 2009.
[2] R. Borndörfer, "Discrete optimization in public transportation," in *1st Indo-US Symp. on Adv. in Mass Transit and Travel Behaviour Research*.
[3] B. A. Catherine Morency, M. Trepaniera, "Measuring transit use variability with smart-card data," *Transport Policy*, vol. 14, pp. 193–203, 2007.
[4] C. S. Jensen and D. Tiešytė, "TransDB: GPS data management with applications in collective transport," in *Proc. of the 5th Int. Conf. on Mobile and Ubiquitous Systems*, Brussels, Belgium, 2008, pp. 1–6.
[5] M.-P. Kwan and I. Casas, "Gabriel: Gis activity-based travel simulator. activity scheduling in the presence of real-time information," *GeoInformatica*, vol. 10, no. 4, pp. 469–493, 2006.