Location Information Retrieval and Querying from GPS logs

Sarunas Vancevicius, 54455053, vsarunas@gmail.com, CASE4

December 19, 2007

Abstract

In this report we'll give examples of existing location aware applications which shows us the location aware computing is evolving to becoming mainstream. Various methods are being developed which are not based on GPS. All of these techniques are being incorporated to applications. The location detection algorithms are important back-end part of this field. In this paper, we'll give an example how the GPS data could be collected, stored, present and analyse the three main approaches to the extraction of significant locations from GPS logs and finally present an interface that could be used to query this data.

1 Overview

Location aware devices are becoming more common, various methods have been developed such as Placelab [1] which uses a database of WiFi access points Beacons to determine approximate location; determining approximate location from GSM cell tower IDs have been successfully incorporated to a mobile phone application, Google Maps with My Location [2]. There are methods which combine various combinations of these. Hightower et al. [3], describe an affective and accurate BeaconPrint algorithm which uses GSM Cell Tower IDs and WiFi Beacons as Placelab. These methods have been developed to increase the accuracy of GPS based solutions, and to avoid the problems with GPS solutions. The main problem with GPS solutions is GPS sometimes reports inaccurate location(noise) and the signal is hard to acquire indoors and urban canyons, areas between high rise buildings. lvarez et al. [4] reported that in a city with no skyscrapers the effect was observed on narrow roads.

There are multiple examples of location aware applications using various methods mentioned:

- Previously mentioned, Google Maps with My Location [2] using GSM cell tower IDs.
- Jaiku[7], a microblogging service based on ContextPhone[8], has a Nokia S60 which application allows friends to share location(based on GSM cell tower IDs) and availability. Jaiku was acquired by Google in October 2007.
- Marmasse and Schmandt[6], describe ComMotion, a location aware to do list based on GPS. The limited signal indoors is exploited in their method. Anytime signal is lost for any significant amount of time, the location must be a significant to the user.
- Lambert[9] describes a diary type application which incorporates locations visited.
- Kang et al. [12] give an example use for a mobile phone to turn on silent mode where the ring is inappropriate like lectures, meetings, cinema.

In this paper we'll describe:

- How the GPS data is going to be collected.
- Describe the current methods to extract significant places and analyse them.
- How this system can be implemented and evaluated.

2 Architecture

In this section we'll describe functional description of the system, outlining the system architecture, analysing the algorithms.

2.1 Collection of Data

The GPS data should be logged on a device which is fairly portable, has plenty of storage for logs and has a good battery life. Previous similar projects have been using mobile phone with ContextPhone software[8] or custom software, or just a laptop with a large battery which lasts for 16 hours[3].

The laptop with large battery has a long battery life, but is bulky. Mobile phones do not offer good battery life, Nokia N95 lasts few hours - half a day if heavily used.

Figure 1: Nokia 770

The device the author will be using is a Nokia 770 Internet Tablet[10] with Bluetooth GPS. In author's experience, running an application which queries the GPS device and scans the Bluetooth environment, the battery would last a full day if the device would not be used otherwise. The limiting factor is the Bluetooth GPS module, which has a battery life of around 7-8 hours.

The data from GPS device is logged on the Nokia tablet in an SQLite database.

An example of data :

The time interval the coordinates are taken varies, the program is set to sleep for 10 seconds before taking a new measurement, but due to the time it takes to establish a connection to the Bluetooth GPS device(sometimes due to errors), the actual time the reading is taken varies.

Time	Date	Lat	Lon	Sat	Speed
11:25:39	2007-12-12	53.38558	-6.25693	03	6.0
11:25:50	2007-12-12	53.38558	-6.25695	03	0.0
11:26:02	2007-12-12	53.38559	-6.25690	03	0.0
11:26:56	2007-12-12	53.38576	-6.25681	03	0.0
11:27:07	2007-12-12	53.38588	-6.25715	04	6.0
11:27:18	2007-12-12	53.38597	-6.25681	04	0.0
11:27:32	2007-12-12	53.38602	-6.25715	04	0.0
11:27:44	2007-12-12	53.38589	-6.25715	04	0.0

Table 1: Sample GPS log data

2.2 Finding significant locations

The extraction of significant locations from GPS data is a clustering problem, the solutions to this problem can be categorised to Partition, Time-based and Density-based clustering.

2.2.1 Partition Clustering

K-Means works by "partitioning the input points into k initial sets, either at random or using some heuristic data. It then calculates the mean point, or centroid, of each set. It constructs a new partition by associating each point with the closest centroid. Then the centroids are recalculated for the new clusters, and algorithm repeated by alternate application of these two steps until convergence, which is obtained when the points no longer switch clusters (or alternatively centroids are no longer changed)."Wikipedia[11]. Ashbrook and Starner[5] have used K-Means algorithm. K-Means has the following disadvantages:

- Not very suitable for extracting locations from GPS data due to requirement of number of clusters before hand.
- The centroids are chosen at random, the generated clusters will be different each time.

- The noisy coordinates are not filtered and thus affecting the final cluster, because of this K-Means favours symmetrical shapes

Because of the above problems with K-Means, many algorithms have been developed which do not have those disadvantages.

2.2.2 Time-based Clustering

To combat the problems with K-Means, Kang et al. [12] have developed an algorithm "which clusters the stream of location coordinates along the time axis and drops the smaller clusters where little time is spent. Specifically, compare each incoming coordinate with previous coordinates in the current cluster; if the stream of coordinates moves away from the current cluster then we form a new one.". In another words, a significant location is which any significant time is spent(time threshold), if the coordinates become separated by a certain distance, then a new significant location is created.

This approach has the following advantages:

- Able to run on low processing power devices real time, detected locations as they happen, not in the batch run.
- Can be fairly accurate if the time and distance thresholds are high. They have found that with large time threshold, 30min and distance, 300m it is possible to achieve precision of 15/19 and recall 15/16.
- Can detect various shapes.

The main disadvantages are:

- Does not take an account of historical locations, for example if a significant place such as a coffee shop is visited daily, but the time spent is below threshold; it will not be detected.
- The locations of Wi-Fi access points was used in the original paper, it is possible that using this algorithm on GPS data will produce different results, the Wi-Fi access points have a single coordinate, thus have no noise, where GPS does.

Hariharan and Toyama[13] have used a similar time-based approached as Kang et al.

2.2.3 Density-based Clustering

Density-based clustering algorithms require 2 predefined inputs, R, the radius of a circle and M, minimum points required in the circle. The way this algorithm works is by trying to select a point from all the points, then it tries to match at least M points within the radius of the selected point, if the minimum point requirement is satisfied a significant place is found. If there are two clusters which share a point between them, that cluster is merged into one. This process is repeated for all the possible points.

Most famous Density-based algorithm is DBSCAN[14]. Zhou et al. [15][16] developed DJ-Cluster to improve the limitations of DBSCAN, mainly sensitiveness to input parameters and big memory usage on large data sets.

The advantages of DJ-Cluster:

- Not as sensitive to the input parameters as DBSCAN.
- Can handle various shapes.
- The inputs can be pre-processed to eliminate various noise - like removing coordinates which were generated during movement, taking advantage of the speed attribute provided by the GPS device.
- The noise which is not removed does not cause the detected cluster to reshape, as in K-Means.

The disadvantages of DJ-Cluster are:

- Its batch run, not a time-based interactive algorithm, thus it is not possible to detect location changes as they happen, thus not really appropriate for mobile devices.

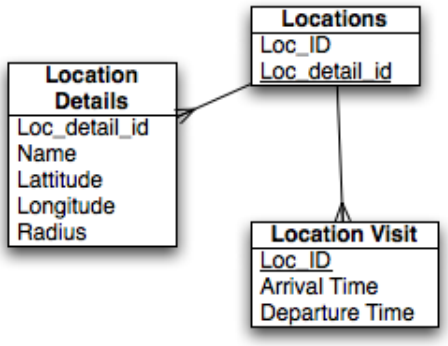

Figure 2: Database tables

- Prior pre-processing of the data is required, otherwise it would be slow; over 3000 points would be collected in a day!

It would be possible to combine the Time-based algorithm and DJ-Cluster to try to use the output of Time-based algorithm as the input to DJ-Cluster.

2.3 Storage of analysed data

Once the locations are retrieved from the data using one of the algorithms described, the data should be stored in normalised database tables as shown in Figure 2.

2.4 Data Querying

There is few possible ways to query the analysed data. The most obvious is to make a simple interface which gives the an interface to query the data powered by SQL commands, i.e defining a set of operations that can be asked:

- When was the last time I was at location X?
- List all the times I was at location X.
- What location do I go after/before location X?
- What is the most common days/hours I go to location X?

All these queries are is just some input for the user, which forms an SQL statement, then the results get presented back to the user. A similar system likes this was implemented by Lambert [9].

Another approach would be to get as much information extracted from the location and get a a text information retrieval system to search it. For example of possible things to expand on each location visited:

- Using reverse geocoding service like GeoNames[17] to get an area name, from this work out what region, city, country it is in. It is possible to even index GeoNames related Wikipedia articles for that location.
- From the date, it is possible to tell the day, month, year, season, is this date any special date(Halloween, Christmas, etc. The system could be expanded to use the dates of contacts and their birthdays) Combining date and location it is possible to tell the weather.

Two very different methods have been defined on how the location data that was extracted from the GPS logs could be searched.

3 Implementation and Evaluation

To implement this system, the data should be collected as described. Then an algorithm based on the ones described should be implemented.

The evaluation is more complicated, and is similar to the way existing information retrieval systems are evaluated. The user who is collecting data should write down the locations he has visited, a standard corpus that can be later reused. The algorithm is then run on this data and the outputed locations are compared to the real locations that have been visited, algorithm input values can be tweaked to get the best results. Zhou et al. [15] describe this method.

This report was done in LaTeX.

References

[1] A LaMarca, Y Chawathe, S Consolvo, J Hightower, I Smith, J Scott, T Sohn, J Howard, J Hughes, F Potter, J Tabert, P Powledge, G Borriello, B Schilit. Place Lab: Device Positioning Using Radio Beacons in the Wild, In proceedings of Pervasive 2005, Munich, Germany.

[2] Google, Google Mobile Maps with My Location, http://google.com/gmm/mylocation.html, Accessed December 2007

[3] J Hightower, S Consolvo, A LaMarca, I Smith, J Hughes. Learning and Recognizing the Places We Go, pp. 159-176, UbiComp 2005: Ubiquitous Computing, Tokyo, Japan

[4] JA lvarez, JA Ortega, L Gonzlez, F Velasco, FJ Cuberos. Where do we go? OnTheWay: A prediction system for spatial locations, 2006, Proceedings of the I International Conference on Ubiquitous Computing: Applications, Technology and Social Issues, Madrid, Spain

[5] D Ashbrook, T Starner. Using GPS to learn significant locations and predict movement across multiple users, Personal and Ubiquitous Computing, Volume 7, Number 5 / October, 2003

[6] N Marmasse, C Schmandt. Location-Aware Information Delivery with ComMotion, 2000, Handheld and Ubiquitous Computing: Second International Symposium, pp. 361-370, Bristol, UK,

[7] Jaiku, Jaiku on Nokia S60 phone, http://jaiku.com/tour/3, Accessed December 2007.

[8] M Raento, A Oulasvirta, R Petit, H Toivonen. ContextPhone: a prototyping platform for context-aware mobile applications, Pervasive Computing, IEEE Volume 4, Issue 2, Jan.-March 2005 pp. 51 - 59

[9] MJ Lambert, Visualizing and analyzing human-centered data streams, May 2005, Masters Thesis, Massachusetts Institute of Technology. Dept. of Electrical Engineering and Computer Science.

[10] JH Kang, W Welbourne, B Stewart, G Borriello. Extracting places from traces of locations, ACM SIGMOBILE Mobile Computing and Communications Review, Volume 9 , Issue 3 (July 2005)

[11] Nokia, Nokia 770 Internet Tablet, http://europe.nokia.com/A4145104, Accessed December 2007

[12] Wikipedia, K-means algorithm, http://en.wikipedia.org/wiki/K-means_algorithm, Accessed December 2007

[13] R Hariharan, K Toyama. Project Lachesis: Parsing and Modeling Location Histories.

[14] M Ester, HP Kreigel, J Sander, X Xu, A Density-Based Algorothim for Discovering Clusters in Large Spatial Databases with Noise, Published in Proceedings of 2nd Internation Conference on Knowledge Discovery and Data Mining(KDD-96)

[15] C Zhou, D Frankowski, P Ludford, S Shekhar, L Terveen. Discovering Personal Gazetteers: An Interactive Clustering Approach, 2004, ACM international workshop on Geographic information systems, pp. 266-273, Washington DC, USA.

[16] C Zhou, D Frankowski, P Ludford, S Shekar, L Terveen. Discovering Personally Meaningful Places: An Interactive Clustering Approach In ACM Transactions on Information Systems (TOIS), Vol. 25, No. 3, Article 12, July 2007

[17] GeoNames, http://www.geonames.org/, Accessed December 2007.