

UC Berkeley

Faculty Research

Title

Overcoming battery life problems of smartphones when creating automated travel diaries

Permalink

<https://escholarship.org/uc/item/6q59m0hj>

Journal

Travel Behaviour Research: Current Foundations, Future Prospects

Authors

Jariyasunant, Jerald

Sengupta, Raja

Walker, Joan

Publication Date

2014-04-01

University of California Transportation Center
UCTC-FR-2014-05

**Overcoming battery life problems of smartphones when creating
automated travel diaries**

Jerald Jariyasunant, Raja Sengupta, and Joan Walker
UC Berkeley
April 2014

DISCLAIMER

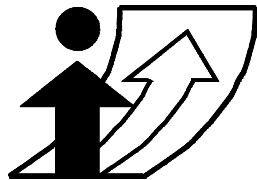
The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This University of California Transportation Center document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program and California Department of Transportation, in the interest of information exchange. The U.S. Government and State of California assume no liability for the contents or use thereof.

Overcoming battery life problems of smartphones when creating automated travel diaries

Jerald Jariyasunant, UC Berkeley

Raja Sengupta, UC Berkeley

Joan Walker, UC Berkeley



**Travel Behaviour Research: Current Foundations,
Future Prospects**

13th International Conference on Travel Behaviour Research
Toronto 15-20, July 2012

Overcoming battery life problems of smartphones when creating automated travel diaries

Jerald Jariyasunant
621 Sutardja Dai Hall
562-412-3729
jjariyas@berkeley.edu

Abstract

Gathering data using travel diaries has been a requirement to analyze travel behaviour data for decades. Traditional pen and paper techniques of gathering data has been complemented by new technologies, most recently GPS loggers, and now smartphones. Smartphones appear to be the silver bullet in collecting travel data on a wide scale, they are widespread, applications are easy to distribute, and the hardware is reliable. However, conserving the battery life in smartphones is a challenge to creating automated travel diaries. The smartphone is not solely a travel diary tool and for a system to work, the travel diary application cannot be a burden to the user. In this paper we leverage work from the “everyday location monitoring” research community in a system to sparsely collect location data from smartphones in a battery efficient manner. With this sparse data, we present an algorithm to generate trips, and suggest six metrics to analyze the quality of the trip determination system. We show how the algorithm has been adapted to deal with problems encountered in the real world (i.e. inaccurate location readings, different capabilities of various phones, various user behaviours turning phone services on/off), with a test of 125 users across 20 states and 23 different models of smartphones. We also evaluate the system across the six defined metrics with a three-month dataset from six users. The results show the accuracy of the automated system, and how the results are significantly boosted with just a small amount of human input. The results show that there exists a trade-off between battery consumption and trip inference accuracy, and the metrics used in this paper are basic measures to be used to determine the quality of an automated trip diary system.

Keywords

Smartphones, trip determination, battery life, travel diary

Preferred Citation

Bibtex

1 ADVANCEMENTS IN AUTOMATED TRIP DIARIES WITH SMARTPHONES AND GPS

Collecting data on the way people travel has been a longstanding research problem; the information is incredibly valuable, can be applied in a variety of industries and research fields, but obtaining a rich dataset is not simple. In the transportation community, work on travel survey innovation has been researched for years, and today behavior modification programs such as travel feedback programs is a growing research topic.

Travel surveys and data collection are essential for travel behavioral studies, which are used to understand individual travel behavior, collecting socio-economic and demographic data, household and vehicle data, alongside travel diaries listing all trips the individual made in a day. The data is used to estimate transportation planning models, which forecast traffic, land use patterns, changes to the transportation infrastructure, and policies. The problem with the current state of practice is that household travel surveys are expensive and do not capture people's lifestyles [1] due to the short durations of the studies. These traditional data collection methods with paper surveys are expensive and require survey subjects to constantly record data, disrupting their normal behavior. Therefore, these surveys are typically only carried out with small sample sizes, at intervals of 10 – 15 years, covering 1 – 2 days of travel. With these sparse amounts of data, models are derived for 20 to 25 year forecasts of both transportation and land use conditions, making strong assumptions to cover for the lack of long-term data (e.g., that work location follows residential location choice). As an example, the San Francisco Bay Area travel survey (BATS) was last conducted in 2000 and only covers two days per study participant in 15,000 households and is used for long-range transportation planning in the entire Bay Area. Researchers have recognized that this style of data collection: running 48-hour surveys with a large sample population does not provide as much value as travel diaries over longer survey periods but with small sample sizes[2]. Because of these difficulties of running surveys, there is a limited amount of data available for travel behavior modeling. It has been suggested for some time that addressing the data collection pain-point is the role that such technology can play [3, 4] .

For decades, researchers have been using GPS technology to supplement travel diaries, and have shown the benefits of using GPS location data to create travel surveys as a result of the increased accuracy. Many early studies focused on using independent GPS devices [5, 6] and validating conventional diary surveys with these devices. These studies started with participants in the range of 100 persons and have expanded to as many as 1500 with the French National Travel Survey and 2200 households in Halifax [7].

Starting in 2007, researchers began looking at smartphones as a potential tool to perform travel survey studies. There are many advantages of using a smartphone versus a separate GPS device. First, the smartphone is a ubiquitous device used by over 200 million Americans. Of the 450 million mobile phones used everyday, 46.8% are smartphones [8]. This number is expected to grow to 70% by 2015. Thus, running

a large-scale travel survey is much more feasible and affordable for researchers and local governments. Another advantage is the distribution of travel surveys - instead of GPS devices that have to be sent to people, installing an application on a smartphone is quick and cheap, and more importantly it is a concept that people are already accustomed with.

Unfortunately, smartphones cannot collect location data in the same way wearable GPS devices collect data because they were not designed for that purpose. Smartphones cannot constantly collect location data due to the power usage of location based services and the limited battery capacity of phones; a battery will be drained in 5 hours if the phone's GPS sensor continuously collects location points[9]. Between running apps, surfing the internet, writing emails, and talking on the phone, there is a limited power budget that a smartphone can use for the purpose of collecting location data for travel studies. On top of that, people are very concerned about the battery life of their phones, are aware of the power consumption of their apps, and actively manage applications running on their phone to minimize battery drain [10]. People have regular charging patterns, and it is the opinion of the authors that any system for creating automated travel diaries must do so in a battery efficient manner that does not change the user's behavior and interaction with their smartphone.

There is a solution: the battery usage of phones can be managed by sampling various sensors on the phone very infrequently. By doing this, it is possible to collect a small amount of location points without draining the user's battery, and still recreate trips using as little data as possible from various sensors: GPS, WiFi radio, and accelerometer. With sparse data, determining trip attributes, even as simple as the start time and location of trips become more complex.

Thus, the challenges of building a travel behavior data collection system is thus two-fold. First, developing an application which gathers location data in a battery efficient manner, and second, developing algorithms to translate sparse location data into accurate trips. Fortunately, there exists a large community of researchers focused on solving the location tracking problem in a battery efficient manner. In fact, many of these researchers had already nearly solved the problem of collecting location data that can be applied for the purpose of generating trip diaries.

Using the ideas developed by this community, a smartphone app was developed to track a user's movements and locations. The smartphone app sends data to a server which runs a trip determination algorithm that outputs the following details of a person's trips taken daily: trip start/end times, start/end locations, route taken, and travel mode used. The trip data is displayed on a web site or can be downloaded to analyze and run travel models on.

In theory, solving this automated travel diary problem is not complex - if phones behave as expected and provide sparse, yet accurate data, and if users never shut off their phones or turn off location services, this research simply combines the work of the battery research community with the research done to determine travel mode on smartphones. However, in practice this is not the case. The number of external variables: types of phones, geographic locations, topographies, user behaviors, users' phone settings, and many more, require a robust solution to the problem.

To learn about these externalities, a lengthy evaluation of the system was con-

ducted, with 125 users providing data on 21 different phones in 20 different states featuring many types of topographies - amounting to a total of over 120,000 user-hours of data recorded. The data provided insight into the issues of various phones, the effect of different types of geographies on the accuracy of data, and smartphone habits of users which affected the quality of the location data. In addition, the quality of the trip determination system was evaluated: a set of metrics is proposed as the standard to evaluate any automated travel diary system, and a detailed evaluation of 1850 trips made over 3 months was conducted, consisting of a thorough daily review of trips made.

The paper is outlined as follows: Section 1.1 gives a history of travel studies using GPS devices. Recent work in the field of travel mode determination on smartphones is described in Section 1.2. The vast field of research of battery usage in smartphones is discussed in Section 2. Section 3 describes the architecture of the travel diary system, and the algorithms which process the location data into trips made are described in Section 4. The evaluation of the system is described in Section 5 and Section 6 outlines the research areas needed to take the project further.

1.1 Trip Determination with GPS devices

Starting in the late 1990s, researchers began incorporating GPS devices into travel diary studies[11]. Transportation researchers recognized that pen and paper travel surveys were not providing accurate enough data in an efficient manner[12], and attempted to use new technologies to improve the state of tracking individual travel behavior.

The first GPS enhanced travel diary studies started with tracking driving movements only. A proof of concept in-vehicle GPS logger was first demonstrated by Wagner [13, 14] in 100 household vehicles in Lexington, KY. In this study, any many subsequent studies, a personal digital assistant (PDA) was used as the data recorder. Respondents were asked to enter such information as the identity of the driver of the vehicle, the purpose of the trip, and the identity of other persons accompanying the driver. Further studies were conducted with subjects in the range of 100-200 participants [15, 11]. The GPS devices showed many flaws in the traditional methods of collecting travel data: the standard trip-based CATI survey conducted in the US under-reports travel by about 20–25% [16], short trips lasting 10 minutes or less are likely to be unreported [17].

Portable GPS devices were developed to capture a person’s activity through out the day, not just while driving. Initial devices were bulky, heavy, had issues with battery life [18], but future iterations lead to smaller, lighter, and battery efficient devices. These devices captured all trips, even non-driving trips, and as a result significant research has been made to develop algorithms that automate the identification of trip details, including trip purpose and travel mode [19, 3, 20, 21, 22, 23] . Studies with wearable devices ranged from small studies of 100-200 people to much larger studies of 2,000 people of varying time periods [18, 24].

Many studies required participants to enter information into PDAs or write down their trips in addition to carrying around GPS devices [25, 26, 27]. Participants ex-

perienced fatigue in recording their travel activity, thus some studies were performed with passive devices: users were required to simply carry the device with no data entry needed. The data from the passive device was collected, processed, and only in some studies the user was asked to respond to a prompted recall survey [28, 29, 30, 31, 6, 16]. In these GPS-based prompted recall studies, the data was processed then presented back to the respondents for confirmation and/or correction of derived trip details and the completion of other traditional elements that cannot be derived from GPS data, such as number of travel companions and destination activities. [32]

1.2 Trip Determination with Mobile phones

1.2.1 Prior to smartphones

Before the widespread popularity of smartphones, researchers explored using mobile phones as a data collection tool, using GSM technology[33]. An electronic questionnaire on mobile phone was also used to record the trip purpose, travel mode, and start and end time of the trip. Locations, which were derived from phone communication towers rather than GPS, were successfully used for estimating travel patterns in a region [34]. Building on these studies, more technology solutions were explored, include Bluetooth, WiFi, RFID, and smart-cards as a replacement, or enhancement to location gathered by GPS [35].

In Japan, PHS (personal handyphone systems), which work by gathering locations from base stations, similar to how location via WiFi beacons work today, became popular for doing geo-location in cities [36, 37]. More than 20 case studies using PHS technology with participants of up to 100-300 were conducted in Japan since 2003 [38, 39, 40, 41].

One problem with these studies is that there have been varying levels of validation performed to date on GPS processing software, with a general lack of ground truth datasets available to perform this critical task[11]. Although mobile phone technologies seem to be an obvious method for collecting travel data, only a handful of studies have verified the accuracy of automated processing algorithms on GPS data from mobile phones.

1.2.2 The smartphone era

An increasing number of researchers have worked on the problem of inferring modes of transportation using smartphones, both in real-time[42, 43] as well as via post-processing[44, 45]. This research attempts to differentiate a person walking, biking, driving, taking a bus, or some subset of these five modes. Common to all previous work is the extraction of significant attributes from the raw sensor data, using a training set of data, and building a model with a classification algorithm. One of the earliest systems predicted not only mode of transportation, but destination, and route taken in real-time with a hierarchical Bayesian network[43]. Greater classification accuracy was reached when post-processing GPS data was enhanced with mapping

data[44]. However, compared with the previous works, the highest levels of accuracy were demonstrated by combining GPS and accelerometer data to determine mode of transportation in real-time[42].

Determining the transportation mode cannot be done with the GPS sensor alone - data about the underlying transportation network is required to differentiate driving with transit trips. Stenneth proposed and validated a method which separated walking, cycling, driving, and public transit [46]. One problem that has plagued both standalone GPS devices and smartphones is problems getting GPS signals in certain regions (near high rises). By augmenting GPS-based positioning with network-based localization to the impact of missing GPS signals is reduced and transportation modes can be determined in a more robust manner [47].

However, creating a solution for automated travel diaries using smartphones is not only about determining the mode of transportation a person takes. It is also necessary to determine the time of the trip, route, and trip purpose among other things. A system called CTrack was developed, which recreated trajectories from GSM data only, due to the battery constraints of keeping GPS running constantly - a major issue in the use of smartphones as data collection tools which is discussed in great detail in Section 2. [48].

The reported accuracy of these algorithms is very high - over 90% in nearly every case - which shows that data collection for automated travel diaries can be done. However, these algorithms have not been used in long-term studies to analyze two things:

1. Battery consumption of the application on smartphones
2. Accuracy of determining characteristics trips made, which include more than just travel mode: start/end times, start/end points, route, and trip purpose among a large dataset.

2 CONCERNS ABOUT BATTERY CONSUMPTION OF SMARTPHONES

When collecting data to generate a person's travel history, location-based services on smartphones are required to be turned on. The use of these location-based services is a major limitation: constant tracking with GPS drains smartphones batteries rapidly, allowing only for a few hours of usage, and making users very unhappy. The issue of battery life is of major concern: the smartphone is not solely a travel diary tool and for the system to work the travel diary application cannot be a burden to the user. Our philosophy when designing the system was: If one is to develop a system that uses smartphones as a tool for automated travel diaries, the system cannot change the regular habits of the person using his smartphone.

2.1 Users care about battery life

Battery consumption on devices used for travel surveys is a major issue. Researchers in the travel behavior community have also been concerned with the battery life of standalone GPS devices. Stopher did a test for average battery life of various GPS devices, finding that the average was 22:46 hours [49]. Battery consumption on smartphones is an even more serious issue. Previous research has shown that existing battery interfaces present limited information, and, as a consequence, users develop inaccurate mental models about how the battery discharges and how the remaining battery percentage shown in the interface correlates to application usage [50].

Rahmati[51] coined the term Human-Battery Interaction (HBI) to describe mobile phone users' interaction with their cell phones to manage the battery available. According to a survey they conducted, 80% of users take measures to increase their battery lifetime, and maximizing battery life will continue to be a key concern for users due to the major usability issues involved in this task. Further studies analyzed user charging characteristics and how users consume battery in their devices [52, 51].

According to the study done by Ferriera, users avoided lower battery levels, with the daily average of the lowest battery percentage values being 30%. In addition, the majority of the charging instances occur for a very small period of time (up to thirty minutes) or between one to two hours, which is the average required time to recharge completely a battery. The charging habits were found to be erratic by time of day, but maintained the average battery percentage greater than 30%.

The result of these studies show that it is important for the travel behavior application on the smartphone to not have a noticeable effect on the battery life of the phone and be able to last through out the day or longer.

2.2 Prior research on everyday location monitoring

Outside of the transportation community, many researchers and businesses are also interested in monitoring users' mobility during daily life. Understanding human mobility provides useful information in a variety of fields [45, 53, 54], and as a result there has been significant research in identifying a person's places of interest using smartphones, often called "everyday location monitoring". This research in this field has direct applications to the travel survey community.

For the everyday location monitoring researchers, the major problem that must be solved is minimizing energy consumption of the smartphone while providing accurate location readings. Continuous measurement of a user's position is possible by keeping the GPS on, however, smarter solutions exist to balance the need to obtain GPS data with extending the life of the smartphone battery. First, instead of simply using GPS, Wireless Positioning Systems (WPS) using cell towers and Wi-Fi access points (AP) are technologies used to provide a user's raw coordinates [55, 56]. Also, rather than simply sampling the GPS, or the other location sensors at a constant rate, new algorithms have a variety of strategies to minimize the amount of time the GPS samples location. In nearly all studies, a movement detector is used, either by

using the accelerometer[57, 58, 59, 60] or a combination of sensing WiFi and GSM networks[61]. An adaptive scheduling policy is also used to adjust sampling rates for the various sensors, depending on the objective of the algorithm: to minimize location error[62, 57], minimize energy consumption[63, 59], or maximize place detection. In a few studies, a mobility predictor is also used to estimate times when a person is likely to move, and adapt the sampling rate based on these predictions[62, 63, 61]. These studies have been used to collect significant locations, or points of interest which a person travels to, and similar innovations are used in our design the system to not only detect significant locations, but also the mode of transportation used by individuals. This requires higher sampling rates and levels of location accuracy while a person is in motion.

This prior research shows the problem of tracking and monitoring a person’s meaningful places (points of interest) using the regularity of individual mobility pattern has been solved. Thorough evaluations of the energy consumption of mobile phones and the various sensors on the phone have been done, and innovative algorithms have optimized the performance of the tracking users’ locations. For this research to be used in an automated travel diary system, this work has to be extended in two ways.

1. The algorithms focus on “where people go”. It is need to know “where people go and how and when people got there”. Evaluations on the battery consumption of apps as well as the accuracy of predicted trips is necessary.
2. The travel survey community does regular evaluations with hundreds of people and knows how to correctly sample from household populations. It is not likely that these households will all carry the same types of phones, and use their phones in the same way.

Thus, it is necessary to perform tests on a variety of phones, in a variety of locales, and with various types of users with different phone usage characteristics (turning on and off GPS, WiFi, charging habits, etc.)

3 OVERALL ARCHITECTURE

Before detailing the way the trip determination algorithm works, this section describes the big picture: the overall architecture and data flow of the trip diary system. The design of the data collection system is shown in Figure 1. It consists of three components: the tracking application on smartphones, the server architecture to handle incoming location data and handle data requests, and the analytics software to transform the raw data into trips made and meaningful statistics and information about those trips. The applications running on the participants’ phones collect raw sensor data which is uploaded to cloud-based server and stored in a database. A periodic job reads the raw data, processes it to infer trip origin and destination location and times and determines route and travel mode(biking, walking, driving or taking transit). This also makes use of information from third party sources such as public transit data which are stored on the server. Trips are further augmented with data such as

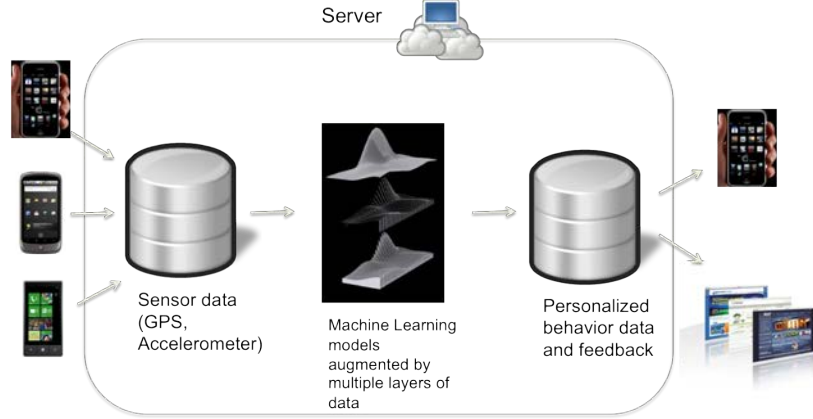


Figure 1: System Architecture Diagram. The components of the system consist of mobile phones, trip determination algorithms running on a server, and web tools to view and correct trips.

addresses/neighborhoods of trips made, distance traveled, time spent traveling, CO₂ emitted, calories expended, travel costs, and other data that can be layered once trip times and locations are determined.

3.1 Mobile application

The mobile application runs in the background on all smartphones running the Android operating system. The goal of the application is to minimize the amount of energy used to collect location and movement data while collecting enough data to allow the algorithms on the server to recreate trips made. In this section, we describe the algorithm implemented on Android phones.

We implemented an algorithm with ideas already implemented by previous groups, as described in Section 2.2. The application used similar techniques to reduce battery consumption: a movement detector using WiFi and accelerometer, an adaptive sensor selection during movement adjusting the duty cycles of GPS, WiFi, and accelerometer, and a mobility predictor based on a person’s familiar locations at various times of day. The only difference between this application and the ones described in Section 2.2, is the adjustment of the duty cycling of the GPS sensor while a user is in motion. In stasis, a progression of sensors are used: initially the accelerometer is used to detect movement, gathering data at a 5Hz rate for 3 seconds once per minute. Once movement is detected, WiFi beacons are used to detect whether or not the movement is a significant location change. Locations are gathered once per minute from WiFi + cell towers once significant location change is detected. Once a person is detected as moving faster than a walking speed, GPS is sampled once per minute, until the person begins to move at walking speed or slower. Thus, the energy consumption of the application can be broken down into two numbers representing the energy

consumption while one is in motion and while one is in stasis. Based on readings from 125 users with 21 different types of phones (described in Section 5.1) on average, the amount of power consumed while a person is in motion is 8.7mA while the amount of power consumed while a person is in stasis is 0.7mA. These values are directly queried from the Android SDK. On average, the application accounts for 3% of total battery consumption at any given time. With the average mobile phone battery capacity being 1500mAH, the energy consumption of the application allows for 33 hours of energy consumption, assuming 2 hours of traveling each day and average smartphone usage (internet use, texting, talking on phone, using apps). This is significantly longer than many applications used in practice, simply collecting GPS data every second, or even every 5 seconds, which will drain a user’s battery in 5 hours[9].

Due to the requirement to minimize battery consumption, the number of location points generated by the application is very low. The map in Figure 2 shows the number of points received when all sensors are turned on compared to the number of points received using our mobile application. The large rings around the points represent the horizontal accuracy of the location points - red represents GPS points, and green represents network points (retrieved from either WiFi or cell tower positioning). In addition, accelerometer points are only requested once per minute while the user is in motion, and mac addresses of nearby WiFi access points are recorded only when a person transitions from stasis to motion, as detected by the accelerometer. These pieces of data from each sensor is sent to the server with a timestamp, which does the work of sorting through these independent readings and generating trips.

Upon launch of the application, the user is asked two survey questions: the number of days per month they bike, and take transit. The results of the survey are used in the mode determination portion of determining trips made, described in Section 4.5. Afterwards, the application runs in the background. If the application is killed or the phone restarts, the application is brought back up after a short delay. This allows for the phone to record data at all times to determine trips.

3.2 Server

The server is the hub of data where all trip processing occurs. Sensor data from the phone is uploaded once every three hours and split into various databases depending on the type of data: accelerometer readings, wifi access points, battery levels, locations (latitude and longitude) based on cell tower / wifi positioning, locations based on GPS, whether or not various location providers are on, and other data points. All the pieces of data play a different role in re-creating trips made by a user: the algorithm is described in greater detail in Section 4. The server contains databases of third party data used to process trips (i.e. transit, weather data). The server is run on a medium sized instance on Amazon Elastic Compute Cloud (EC2), with API access from the phone handled by NGINX storing data in a Postgres and Mongo database.

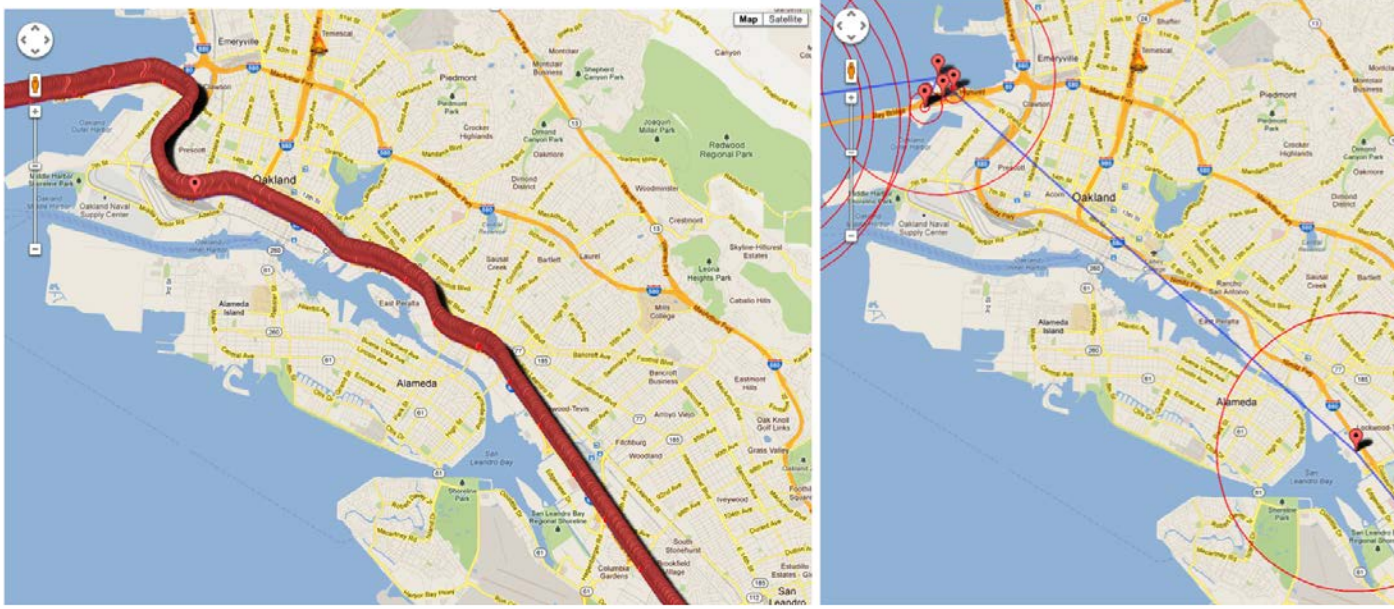


Figure 2: 1Hz data vs. Sparse data. The figure on the left shows the amount of location data from the phone with the GPS sensor gathering data at 1 Hz. This exemplifies the type of data a standalone GPS logger would receive. The figure on the right shows the amount of data generated by our mobile application. The red circles represent the horizontal accuracy values, which captures the uncertainty of the location.

3.3 Website - Online Mapping Tools

A web interface was written to provide a suite of tools to handle research subjects' data. The interface allows a researcher to process trips with one-click, and also allows the researcher to visually inspect the set of trips generated for any user with mapping tools. In addition to simply inspecting the trips for correctness, the researcher can correct the trip, by selecting different start/end points, changing the start/end time, adjusting the route taken, or transportation mode.

These tools are used in the evaluation of the algorithm, described in Section 5. Using the tools, testers validated the accuracy of their trips by viewing their data and having the researcher correct the computer generated trips; the users' inputs generated the ground truth which was used to evaluate the algorithms, as shown in Figure 3.

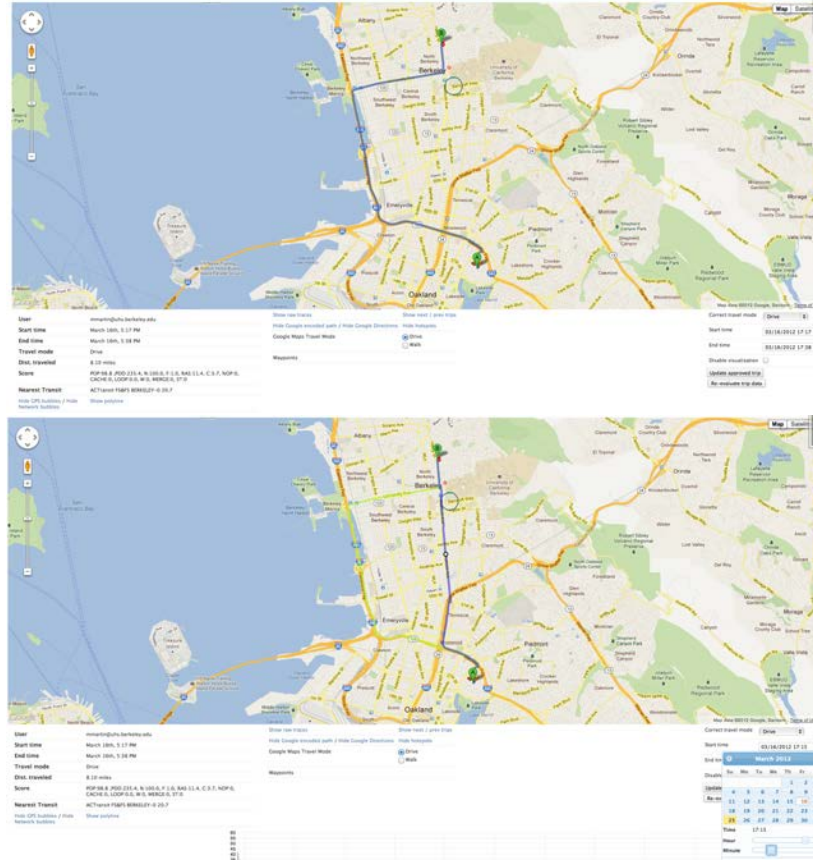


Figure 3: An example of the tool used to correct a trip’s route and start time. The figure on top shows the trip data that is generated by the trip determination algorithm. The figure on the bottom shows the trip after the route and start time have been corrected.

4 AN ALGORITHM FOR TRIP DETERMINATION WITH SPARSE DATA

The goal of the trip determination algorithm is to take the sparse data from the phone’s sensors and recreate trips made. For the purpose of this paper, the definition of a trip is a period of motion surrounded by two periods of stasis greater than 3 minutes. During this period of motion, a person must have traveled a minimum of 250m from an origin to destination able to be mapped to points along the street network. As described in Section 3.1, and shown in Figure 2 data from the location sensors is not frequently generated, however, the much bigger issue is that there exists a wide variety of phones that have a variety of behaviors which can only be learned through real world testing. A list of some of the problems encountered during testing of the application are described in Section 5.1. These discoveries from field testing the application have contributed to the various processes which the data flows through. A flowchart of the trip determination algorithm is shown in Figure 4, and each block

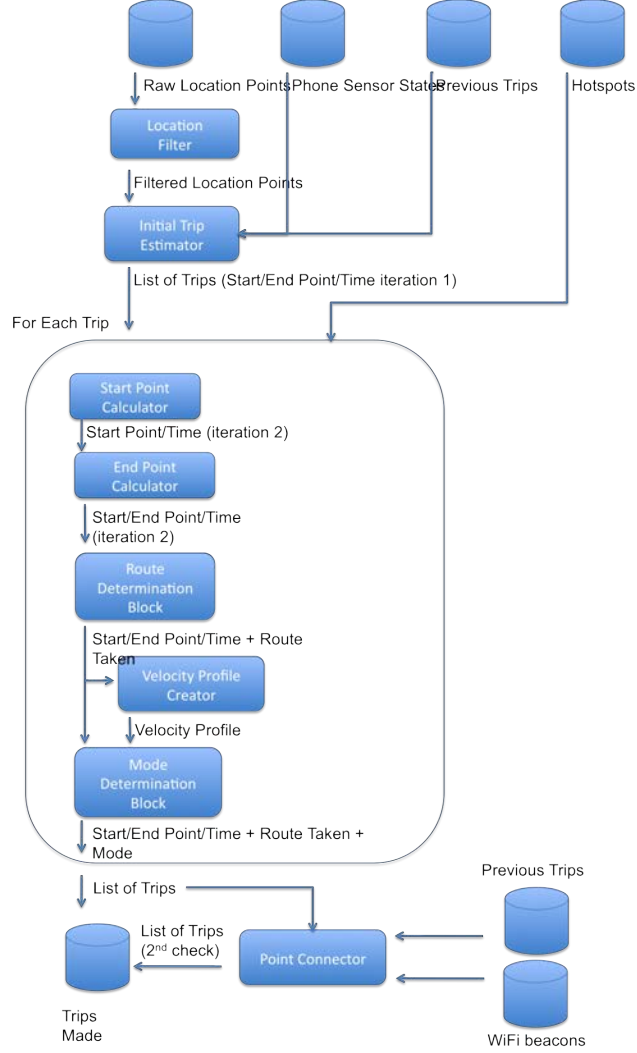


Figure 4: Overall algorithm flowchart. Each block and data source is described in Sections 4.1 - 4.6

in the flowchart is described in the following sections.

4.1 Filtering noisy data

All location points generated by the phone are returned with a latitude, longitude, horizontal accuracy (the error range of the latitude longitude pair), velocity, altitude, and the source of the location point (from GPS or Wifi/Cell towers).

In theory, the true location of the phone is contained within the centroid around the latitude, longitude geopoint with value of the horizontal accuracy as the radius of the circle. However, this is not true; once in a while completely inaccurate GPS points are created, and inaccurate network points occur extremely frequently. In a

test of 777,670 data points gathered from the data set of 125 users (described in Section 5.1), under 1% of GPS points lied outside the circle defined by the latitude, longitude and horizontal accuracy, while 51% of Network points, with the location sourced from either cell towers, or WiFi beacons lied outside the circle.

Despite the fact that network points are often inaccurate, they are used as part of the mobility detection algorithm, to ensure that the GPS sensor is not used very frequently. Thus, faulty points are simply filtered out before attempting to build trips. Although much more rare, inaccurate GPS points are generated from time to time. Like the inaccuracies with the network points, the reasons for this are unknown, but the points are filtered out as well.

Location points are filtered under the following conditions:

1) If two points are generated with the same timestamp, the more accurate point is taken.

2) If the speed required to travel the distance between the edge of the location centroids is greater than 105 miles an hour between consecutive points, but the first and last point is less than 105 miles per hour, the in-between points are removed. The speed of 105 mph was chosen from empirical evidence of the same observations used to calculate the location accuracy values described in this section.

3) It has been observed that once in a while the GPS sensor can enter a “disrupted mode” in which it can temporarily return faulty points outside the horizontal accuracy centroid for up to 2 minutes at a time (based on empirical evidence). Any points created during the “disrupted mode” are removed.

4) Faulty networks points often are returned repeatedly from the network listener - if a WiFi/Cell based location point is tagged as faulty even once, all future instances of the exact same location point are removed.

The filtering algorithm with the preceding rules is run forwards and backwards in time.

4.2 Hotspots

Before any trips are calculated, the raw location points are used to identify locations where a person spends a large amount of time (i.e. home, work, gym) which we refer to as “hotspots”. In Section 3.1, we referred to a number of papers which minimized battery life to identify points of interest. We follow the same logic here, and further describe the methods for identifying trips made in the following sections - the contribution of this paper on top of the research from Section 2.2.

Figure 5 shows the algorithm flowchart for hotspot generation. The first step, the location filter, is used such that all non-GPS points and GPS points with accuracy greater than 66 meters are not used to determine the hotspot centers. In the process of marking these points, each GPS point with a horizontal accuracy of under 66 meters is assigned a duration for which it can be proven that the person was at the latitude, longitude specified by that GPS point.

The duration of each point is set as follows. For each GPS point, each subsequent location points are analyzed to see if it can be proven that the person has moved. First, if the areas covered by the centroids (specified by latitude, longitude, and horizontal

accuracy) of the location points do not intersect, then the duration of the GPS point is set as the difference in time between the two points. In addition, WiFi beacon scans can prove a person has moved. If the intersection of access points seen by two subsequent WiFi beacon scans is null, then the duration is set as the difference in time from the second WiFi beacon scan and the time the GPS point was captured. Figure 6 illustrates a situation in which location points are used to set the duration which the hotspot is valid. In the figure, the duration of the points in the hotspot cluster is calculated from the time of the earliest location point seen in the cluster until the time of the earliest location points seen in the other cluster. The points in the other cluster may be as a result of movement, or faulty readings from the phone, but because neither of those can be proven from the raw data itself, it is assumed that the person has moved.

The output of the location filter is a set of latitude-longitude-duration tuples, which is run through a weighted clustering algorithm[64]. Any hotspots within 250m of each other are merged, with the center of the hotspot set as the hotspot with points holding the longest duration values. Any hotspots with a total duration value of less than 2 hours is eliminated - this is an arbitrary value used to capture a person's total points of interest. For example, if a person spends two hours at a library, or goes to a fast food restaurant for 20 minutes at least 6 times, both will be considered hotspots. However, a location where a person repeatedly drops off or picks up another passenger probably would not be captured by this algorithm.

Finally, each hotspot is annotated with the set of all WiFi beacon seen within a 250m radius of the point.

4.3 Determining Start/End Location of trips

Determining trips requires input from a variety of databases, raw locations, phone sensor states, prior trips made, and hotspots. The data from the various databases are fused, as the algorithm makes a series of guesses to determine the features of each trip made.

The first guess for the start and end location of trips initiated by the state of all the sensors on the phone: this is defined as the sampling rate of the accelerometer, WiFi beacon sensor, GPS, and Network Listeners. When the phone detects a person is traveling at a speed faster than a walking speed, the duty cycle of GPS is reduced - the time first instance of the change in duty cycle is the initial guess of the trip start. The nearest location point to that timestamp is defined as the first guess of the trip start location. Likewise, the time in which the duty cycle of the GPS sensor increases back to the stasis mode is the initial guess of the trip end. The nearest location point to that timestamp is defined as the first guess of the trip end location. Based solely on the raw location points and the state of the phone sensors, a set of initial trip guesses are made.

For each initial trip guess, the trips are refined to a second estimate of the actual trip start/end time and location. The second guess of the start location is determined

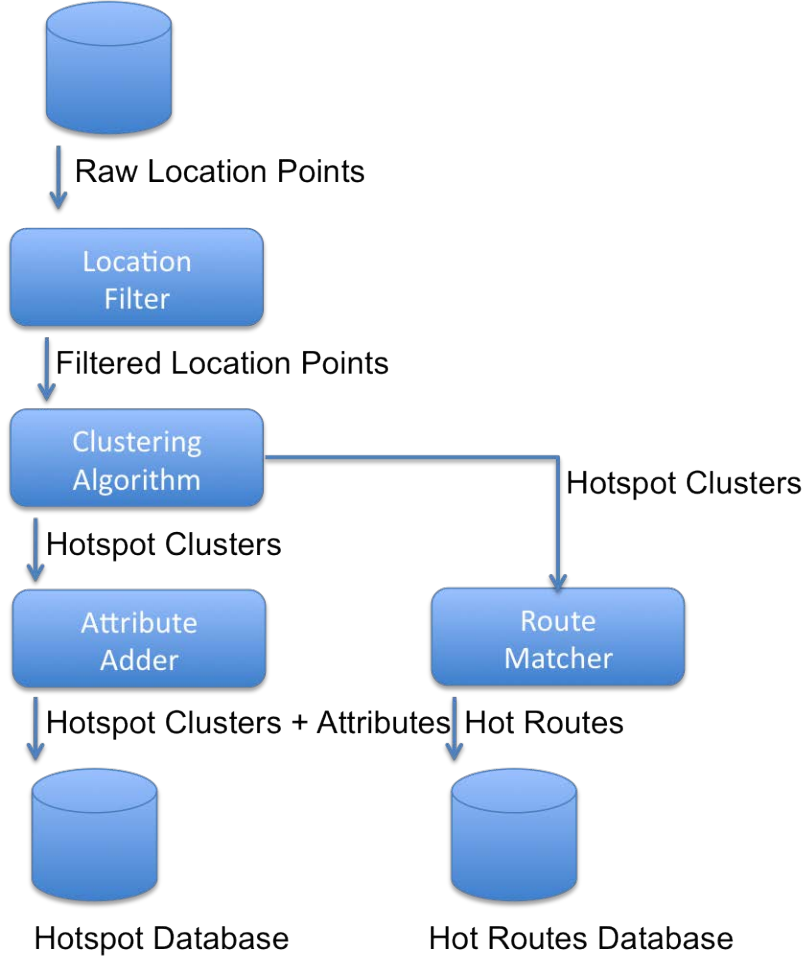


Figure 5: Hotspot algorithm flowchart

by iterating through the filtered location points backwards in time - after it can be proven that the smartphone has not been moving, as described by Section 4.2 and Figure 6, the set of location points during the timeframe in which the phone is in stasis is returned. Each point in this set of points has a corresponding horizontal accuracy; these points are sorted by horizontal accuracy. If the most accurate point in the set has a horizontal accuracy below 66m and is generated by the GPS sensor, it is assigned as the starting point of the trip. Otherwise, if there exists hotspots that are contained within the shape created by the set of potential starting points, those hotspots are set as the potential starting points of the trip. The hotspot which most closely matches the characteristics of the start time of the trip is assigned as the starting point of the trip - for example, if a trip starts at 7:00AM, the hotspot which corresponds with the location where the person usually spends the hours of 6PM - 7AM daily (most likely one's home) is favored over the location where the person spends the hours 3PM - 4PM on Saturdays. If no hotspots are contained within the shape created by the set of potential starting points, the raw location point with the

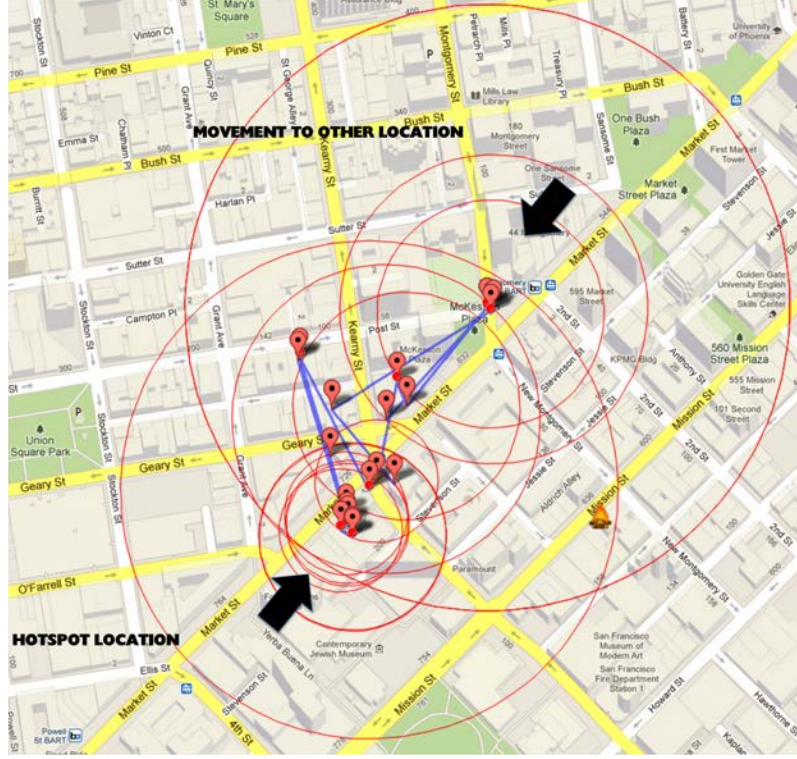


Figure 6: Example of location points at a hotspot center and at another location which signifies movement away from the hotspot cluster.

smallest horizontal accuracy is used - trips which fall into this scenario are rare, and are described in greater detail in the evaluation of the accuracy of hotspots in Section 5.4.1.

The initial guess for the start time of the trip is determined a combination of the accelerometer, Wifi, and location sensors. The accelerometer runs on a duty cycle of approximately 1 minute; when the accelerometer detects movement, the phone scans for WiFi beacons. If the intersection of the set of beacons seen at subsequent WiFi beacon scans is null, then it is assumed that movement occurred during the time between the two scans.

Determining the end location and time of scans is easier than determining the start location of trips. To determine the start time, many of the sensors have long duty cycles in a “low-power” mode. On the other hand, while the person is in motion and arrives at their destination, the sensors have shorter duty cycles. The end location of a trip is determined by iterating through the filtered locations forwards in time from a location prior to the initial guess of the end location. After it can be proven that the smartphone has not been moving, same steps are taken as determining the start point of trips - sorting the potential endpoints and using the hotspots database when necessary.

Determining the end time of a trip is based on a combination of the location sensor and accelerometer. If the accelerometer detects a prolonged period of no motion, the

trip end time is defined as the time of the first reading of the no motion period. However, while the accelerometer can provide a clear signal when the phone is still, not all trips end with the phone being still - for example, after one has arrived at an office building, the person must walk up the stairs to his final destination. If a prolonged period of stasis is not detected, the location sensor is used as the method of determining the trip end. The same set of location points used to determine the end location - proving that a person has stopped moving - is used by using the time of the first point in the set of points as the end time. In the event that accurate location points are not returned from the phone, WiFi scans are used, If a single common access point is seen between two consecutive scans, the end time of the trip is set as the time of the first of the two WiFi scans.

4.4 Determining Route taken

Once the trip's start and end locations are determined, the route taken between the points is generated. The route determination algorithm uses these start points, the raw location points, and the Google Directions API[65]. The flowchart of the Route Determination block is shown in Figure 7.

While it is possible to have built our own routing engine and mapping database, the routing algorithms were outsourced to the Google Directions to save time and take advantage of the expertise of the Google team. The input to the web request are, the starting and ending points, a set of waypoints, a choice of directions by mode (driving, bicycling, walking), and whether or not alternate routes should be searched. The output of the API is a set of shortest paths (by time) between the origin, destination, and that run through the waypoints. It should be noted that there are a couple of minor disadvantages of using Google Directions. First, the number of calls to the API is limited to 2,500 requests per day, with a limit of 100,000 requests for paying customers. Second, the number of waypoints that can be sent to the API is limited to 8, and for paying customers the limit is 23. This means that any trip which uses extremely suboptimal routes(by travel time) such as looping around blocks or taking a long series of residential streets instead of the highway, may not have its routes calculated accurately.

Before sending a request to the Google Directions API, a set of possible waypoints is generated. The purpose of generating a set of waypoints that does not include a raw points timestamped between the start time and end time of the trip is to minimize the chances of generating an incorrect route. The Google Directions API assumes the origin, destination points and waypoints are exact, while the phone returns points with horizontal accuracy readings that are not centered at the exact latitude longitude point. Location points greater than 100m accuracy potentially encapsulate multiple streets, and even GPS points with a 5m horizontal accuracy often are centered on the wrong side of the street, which affects the route returned from Google. Getting the route correct is important not just for the sake of being correct, but the route is also necessary for the map matching of transit routes when determining travel mode, as described in Section 4.5.

To minimize these routing errors, the most accurate GPS point, measured by

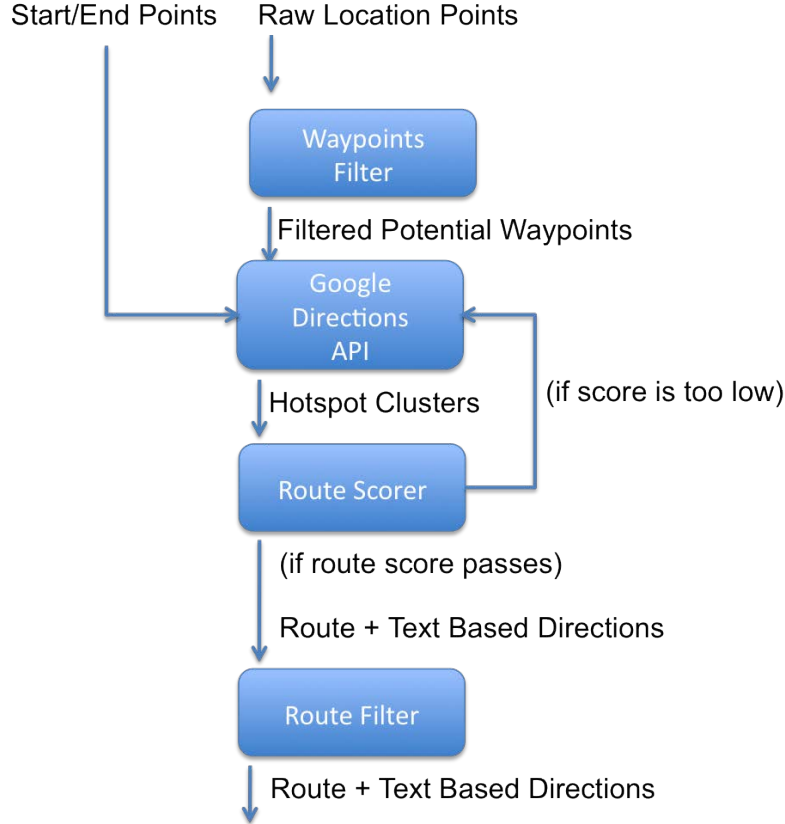


Figure 7: Routing Algorithm flowchart

smallest horizontal accuracy radius, for each minute is kept in the set of waypoints, all other points are filtered out. Before waypoints are considered, only the origin and destination points are sent to Google. One to three routes are returned, and if none of the routes pass through all of the raw points, the path is sent to google again with a waypoint at the most accurate point along the section of the route missed by the returned routes from Google. This process is repeated until the route matches or the waypoint limit is exceeded.

4.5 Determining Mode taken

The last piece of data needed to generate a trip is annotating the mode of transportation to the start/end time and location and route taken. Determining travel mode is done in two steps: a classifier to differentiate between motorized modes, bicycling, and walking, and a map matching algorithm to differentiate between public transit and driving. The travel mode is defined as the main mode a person during a trip (defined in Section 4). While a trip may consist of many segments, such as walking to a car in a parking lot, then driving home, only the main mode, here, driving, is

classified.

4.5.1 Classifier

The classifier is built with the same techniques used in prior research to determine mode of transportation as described in Section 1.2.2. In that research, techniques were developed to perform real-time mode determination, which identifies the transportation mode being used at each unit time interval (e.g. every second). The purpose of this classifier is to identify the main mode used during a trip, but the same steps are followed: collecting a set of training data, identifying a large set of features, performing feature selection, and picking a classification algorithm. The only difference this research and prior work is the set of features selected; instead of using features which can only be observed in real-time, features that are defined over the course of a trip are used, such as the median speed between the trip start and end times.

A set of training data was collected daily across various modes of transportation over the course of 1 month for 6 individuals. A set of features was identified, which included:

1. Speed
2. Trip duration
3. Trip length (distance covered)
4. Expected Driving duration (obtained via Google Directions API)
5. Expected Biking duration (obtained via Google Directions API)
6. Expected Walking duration (obtained via Google Directions API)

Although the accelerometer is a useful sensor to determine transportation mode, features from the accelerometer are not used for the following reasons. Velocity alone can differentiate walking from biking/driving modes. Biking and driving cannot be successfully distinguished because similar accelerometer signatures when the phone is placed in a backpack, or a location that does not allow the phone to capture the biking motion. Previous work has shown that Decision Tree-based classification is simple and effective for classifying the sensory features into transportation modes [42, 44]. In this work we used the Random Forest classifier, which is also tree-based, based on our preliminary experiments and comparisons with Decision Trees. A Random Forest classifier consists of a number of decision trees, each performing classification and voting for the predicted class. Overall output is the class which has the highest number of votes. Each decision tree is built using samples randomly selected from the training set. While growing the tree, a fixed-size subset of features is randomly selected at each node. Random Forests are known to be efficient on large datasets, estimating missing data and handling unbalanced datasets. In addition, they evaluate the importance of variables for classification[66]. In this work, we built a Random Forest with 10 decision trees and 6 attributes.

The classifier predicts between walking, motorized, or biking. An additional classifier was built to predict only between walking and motorized. Based on the results of the survey questions asked in the mobile application about the frequency of biking (Section 3.1), the appropriate classifier is used. For example, a user who responds that he never bikes will have his trips processed through the classifier that distinguishes between walking and motorized only.

4.5.2 Map Matcher

Before the entire trip generation algorithm is run, a database of transit route configurations is built using open transportation data. In 2006, Google established a standard for encoding transit data called GTFS, which contains data that can construct route configurations. Files that store the following relevant information in the structure specified in Figure 3, the route name, direction name, stop / intersection name, latitude, longitude, stop sequence, and agency name. A route configuration is categorized by a route, direction, and agency name. Each configuration specifies a set of stop names with corresponding latitude and longitude points in the sequence which the bus or train traverses the route. Configurations with the same route, direction, and agency name are grouped together. The data also specifies whether or not the path taken by the route occurs on the road network or on dedicated lanes. Currently, GTFS-Data-Exchange[67] is the premier repository of all open transit data stored in the Google Transit Feed Specification format. GTFS files are pulled from GTFS-Data-Exchange and converted into route configurations which are stored in a database. At the time of writing, the database stores data from 339 different transit agencies.

At the time of processing, trips with a route are sent to the Map Matching algorithm to determine the percentage of the route that matches with any bus or train lines which crosses the route taken. The algorithm for performing the map matching is a variation of the Bentley-Ottmann algorithm[68], which finds intersections in set of straight lines. The bus route, and the actual route taken are defined as a series of latitude and longitude points, which when connected form the set of straight lines which the algorithm iterates through. The bus or train line with the highest percentage is returned, and any trip matching greater than 90% is defined as a trip taken as a transit trip.

4.6 Catching all trips made

The preceding Sections 4.2 - 4.5 describe the flow of data through the algorithm when the phone properly adjusts its' duty cycles to capture location data while a person is in motion. However, many trips are made while the phone does not function properly. There are many reasons for this; as discovered in the evaluation in Section 5, users often turn off their phones, turn off GPS or other location services, and most commonly, the phone does not respond properly to requests to return readings from the accelerometer, GPS, or Network listener. To account for these situations when trips are made, but zero, or very few sensor readings are received, data flows through

a separate module.

This module, called the Point Connector in Figure 4, evaluates all sensor data during the “non-trip” times when the person is expected to be in stasis. The same clustering algorithm as the one described in Section 4.2 is used to identify locations where a person potentially could have traveled to. Like the hotspot algorithm, the locations must have a horizontal accuracy reading of 100m or less, due to the number of inaccurate network point readings returned by the phone. In the situation that no point in a cluster has an accuracy of under 100m, sensor readings from WiFi access points are queried against the master WiFi access point database to provide a more accurate position.

The result of the clustering algorithm is a set of centroids defined by latitude, longitude, and horizontal accuracy, and the corresponding time a person was assumed to be at that centroid. The centroids are sorted by time and routes are generated by sending the origin and destination with no waypoints to the Google Directions API. The end time of the trip is set as the timestamp of the earliest reading at the centroid, and the start time of the trip is set as the same timestamp minus the duration of the trip as returned by the Google Directions API. The mode of transportation is determined in two steps. If the same trip has been made in the past, the transportation mode most commonly used for that trip is set as the transportation mode. If not, the transportation mode most commonly used for a trip of that distance is set as the transportation mode. Trips which are generated by this method mainly occur during short trips under 2km and are less accurate in terms of start/end time, route taken, and transportation mode.

5 EVALUATION

Evaluations were performed on the system to accomplish two goals. The first goal was to get the application running on as many different types of phones, living in different locations, with different user behaviors as possible. The second goal of the evaluation is define a set of metrics and to test accuracy of trip determination system with these set of metrics. In this evaluation, separate from the first evaluation, a set of users manually checked the accuracy of all trips generated by the system and made corrections to set the ground truth for trips made. This evaluation consisted of a three-month process in which six users discussed their trips daily with the researcher and used web-based tools to correct their logged trips. This allowed us to evaluate the errors in the start/end time, start/end location, route taken, and travel mode taken for these trips made and determine error statistics, instances in which the system missed trips, or incorrectly created trips when no trip was actually made. These six error metrics are described in the following sections and are proposed as the measures for which all automated travel diary systems should be evaluated upon.

5.1 Phone Tests

This system is meant to be used by researchers to gather data from the field. In the field, there exists a large number of uncontrollable variables: smartphones are made differently, by different manufacturers, users live in different locations which don't produce data in a uniform way across all topographies and locations, and user's phone behaviors are very different. To run a large scale trip diary study, the system must be robust enough to handle all different types of scenarios and corner cases which do not show up in a limited study.

The application was tested with 125 different users on a number of phones in a variety of states and cities with different topologies and population densities. The phones tested include: HTC Desire, HTC Hero, HTC Droid Incredible, HTC Incredible S, HTC Evo 4G, HTC Inspire 4G, HTC myTouch, HTC Wildfire, HTC Thunderbolt, LG LS670 Optimus S, LG P970 Optimus, Google Nexus One, Motorola Atrix 2, Motorola Droid, Motorola Droid 2, Motorola Droid X, Motorola Droid X2, Motorola Droid Pro, Samsung Galaxy S II, Samsung Galaxy Nexus, and Google Nexus S. The data was collected in a variety of cities with different topologies and population densities, in the states of Washington, Oregon, California, Texas, Utah, Illinois, Missouri, Alabama, Georgia, Florida, South Carolina, North Carolina, Virginia, Ohio, Maryland, New Jersey, Massachusetts, New York, New Hampshire, and Ontario, Canada. This data was collected over the course of 9 months for a total of 6,440 days of logged data across all testers. The 125 testers used the application for different periods of time, from as short as 3 days to as long as 9 months, starting in July 2011. 30 of the 125 testers ran the application for more than 3 months.

This evaluation allowed us to learn about the problems encountered from the field, which is divided into phone problems, and user issues. Because of these problems, the phone does not always detect when a person is in motion, and trips are not always perfectly detected.

5.2 Phone issues

Across all phones, it was very quickly learned that location readings from non-GPS sources return inaccurate readings very frequently: this means that the true position of the person lies outside the circle defined by the latitude, longitude, and radius being the horizontal accuracy returned from the phone's location readings. The number of inaccurate location readings was as high as 80% for some users while being as low as 5% for other users. The reason for the inaccuracies was not investigated, but the filtering algorithms described in Section 4.1 were adapted as a result of the large number of inaccurate network readings.

While most phones were found to be able to access the accelerometer in the background, several models of phones are unable to do so, mostly from the Motorola line of phones (at the time of testing running Android 2.3 or lower). As a result, WiFi beacons were necessary to act solely as the motion detector instead of both the accelerometer and WiFi beacon sensor.

The accuracy of location readings were also highly affected by geographic topology. It has been widely reported that GPS does not work well in areas with large skyscrapers, however this study showed location readings in areas with skyscrapers were very accurate, possibly due to both the density of WiFi access points used in positioning. The areas which provided the most inaccurate readings were suburban and rural areas when only location was attempted to be gathered from non-GPS sources. In these scenarios, network points were inaccurate up to 10 miles for one particular user, and were about 1 mile off the actual location in general. Most troubling is that multiple inaccurate points were often returned in sequence, making it appear as if the phone could be moving. In these instances, the accelerometer, and WiFi access point scans allowed for the algorithms to ignore the result of the inaccurate network points.

While inaccurate network points are common, inaccurate GPS readings are very uncommon. However, errors in GPS readings may lead to faulty generation of trips made. It has been observed that GPS readings of high accuracy often drifts anywhere from 20m to 50m while the phone is in stasis. In addition, readings have been observed to be as far as 1500km away from the true location of the phone. While faulty trips are generated, many of these trips do not follow the road network - leading the system to flag the trip as a potentially inaccurate trip.

Finally, it is expected that when the app makes a request for a reading from the GPS, accelerometer or any sensor, the appropriate process will execute to provide the data to the app. However, this is not always the case: 82% of the time when a location point is requested, a point is actually returned from the operating system to the app. The app tries again if a location is not obtained, however, there have been instances when a person has traveled for up to 32 kilometers with the app requesting GPS the entire time but with no location points generated. For the accelerometer: this problem occurs on average 95% of the time. This behavior is often hard to predict and reduces the accuracy of the route taken on a trip.

5.3 User behavior issues

In the Android operating system, a user can choose whether or not to turn on GPS sensors, Network location sensors, and data connection. To perform optimally, the application requires that these three settings are turned on - however during the tests, we did not force the user to do so. A small notice was shown to users that the settings should be, but did not require users to follow the given instructions. This allowed us to observe users' phone behaviors and their affect on the trip diary application running properly.

Not all users turned on GPS services on their phone. It is widely known that turning off GPS increases the battery life of phones, and 4% of the testers followed this advice > 99% of the time. An additional 2% of users frequently switched GPS settings on and off, while the rest constantly kept GPS running. The system still works when GPS services are turned off, however, the start/end locations trips can only be determined to the nearest kilometer for most trips.

Another problem encountered is users turn off their phones, allow the phone to run out of battery, or the phone reboots (for undetermined reasons), although these cases

happen infrequently. In these situations, which occurred .11% of the total recorded time, no data could be collected. In cases when this occurred between trips, the algorithm in Section 4.6 could be implemented, but in cases when round-trips were made, the phone is unable to detect trips.

5.4 Trip evaluation

After the 125 person test was conducted, a second evaluation was run to evaluate the accuracy of the algorithm for detecting trips made. This evaluation consisted of 6 people over 3 months, which generated a total of 1850 trips. Due to the extensive work required to have people manually verify their trips made, a decision was made to run a long test with few people, rather than a short test with a large amount of people. The reason for the smaller number of people in this test compared to the prior test is the ability to gather a dataset with large longitudinal depth to capture people’s lifestyles, hotspots, and travel behaviors. However, the 3 month period is the longest study of its kind for an evaluation of a travel diary system using smartphones. To collect the ground truth for trips made, each of these trips were discussed verbally with the researcher to gather the most accurate readings and manually verified using the tools described in Section 3.3 . This one-on-one discussion made possible the detailed metrics by pinpointing locations of trip start/end locations, hotspots, and routes taken on a map to gather error rates measured to the meter accuracy.

5.4.1 Hotspot evaluation

The first evaluation of the system is the evaluation of the accuracy of the hotspots identified by the system. Validating the accuracy of determining hotspots is important because most trips made start and/or end at a hotspot. Across all users, of the trips made, 48% of trips occurred between two hotspots, 32% of trips occurred starting or ending with a hotspot, and 20% of trips did not include a hotspot at all. The use of hotspots is important because accurate location data at the start/end point is not always present for all trips.

The most common hotspots, such as one’s home and work can be identified in the first day of using the system while other hotspots such as a friend’s house or the supermarket slowly enter the system as they are visited.

5.4.2 Method for verifying accuracy of the data

Here, we propose a definition of a trip, and the 6 metrics used to verify the accuracy of the trip determination algorithms and overall system:

1. Start Time
2. End Time
3. Start Location
4. End Location

Mode	Trips Logged and Verified
Biking	154
Driving	998
Transit	349
Walking	349

Table 1: Mode Split for trip determination accuracy evaluation.

5. Path Match Accuracy

6. Transportation Mode

Of the 1850 trips recorded, the start/end location, was verified for all trips, while the mode and route taken was verified for only 609 trips, which occurred during the last month of the evaluation. The mode split of the trips made is described in Table 1. The start/end time of trips were verified in a different manner, described in Section 5.4.6. The reason why the start/end time, path, and transportation mode were not validated for all trips was because of the fatigue of manually correcting trips daily by the testers.

5.4.3 Errors in Origin and Destination location

Errors in location are measured by the true start/end location of the trip as validated by the person making the trip compared to the start/end location automatically generated by the system. The true start/end location was recorded using the online web tools described in Section 3.3 which used a draggable google maps marker to set the location. The average error for a hotspot location was 25m with a standard deviation of 101m. Across all users, a total of 182 hotspots were detected. This error understandably increases to 197m with a standard deviation of 599m for origin/destination locations which were not hotspots.

There results show that errors in hotspot trips is significantly lower than trips that do not contain any hotspots. Across the 3 month evaluation, 90% of hotspots were discovered by the 3rd week for 4 out of 6 of the users. Two users moved their residences during this time, thus their hotspots changed significantly before and after the move. These results are promising for long-term studies, but a wider evaluation with more users is required to validate this.

5.4.4 Routing Errors

To measure errors in the route generated by the system, the system was evaluated with three measures: meters traveled in which predicted route matches true route, meters traveled in which predicted route does not match any part of the true route, meters of the true route which is not covered by the predicted route. These three measures were translated into metrics called :

1. Route Match : meters traveled in which predicted route matches true route / true route meters traveled, expressed as a percentage
2. Route Error 1: meters traveled in which predicted route does not match any part of the true route/ true route meters traveled, expressed as a percentage
3. Route Error 2: meters of the true route which is not covered by the predicted route / true route meters traveled, expressed as a percentage

Figure 8 shows an example of the three different types of errors. In the evaluation, only driving, transit and biking trips were evaluated. Many walks occur in parks, on campus, or areas where there is no mapping data for all paths.

Table 2 shows the breakdown of errors across these three metrics for the six individual users. The errors are separated by user to show the larger errors in the system for users who use underground transit systems. Users 1 and 2 primarily travel by underground subway, which does not run on a path able to be mapped by the Google Directions API, which only returns results for driving, bicycling, and walking directions. On the other hand, users 3 and 4 use the underground subway once in a while, and users 5 and 6 are primarily auto users.

The second half of Table 2 shows the routing errors after user input. Each day, after the trips were generated, users corrected their trip details: start location and end location, and for one month users corrected their start location, end location, route taken, and transportation mode taken. To generate data for the second table, the trip determination algorithm was re-run daily, taking these corrected routes, called “hot routes”, as generated by the hotspot algorithm described in Section 4.2. These “hot routes” were used only for trips between two hotspots, and significantly increased the route match percentage. The evaluation shows that with a set of user input, there is potential for designing a feedback system to greatly increase the accuracy of the overall trip determination system.

5.4.5 Travel Mode Errors

The errors in the travel mode determined by the system are described by the confusion matrix in Table 3.

The errors show that much work is needed to improve the system. Reasons for the errors include the following:

1. One of the users is a very fast biker in the city, and the set of features used in the classifier did not distinguish between motorized and biking very well.
2. Because some transit trips are made underground, location points had very large horizontal accuracy values, causing the route to be calculated incorrectly
3. Errors in the route predicted by the system propagated to the travel mode determination due to the map matching of incorrect routes.

ROUTING ERROR 1

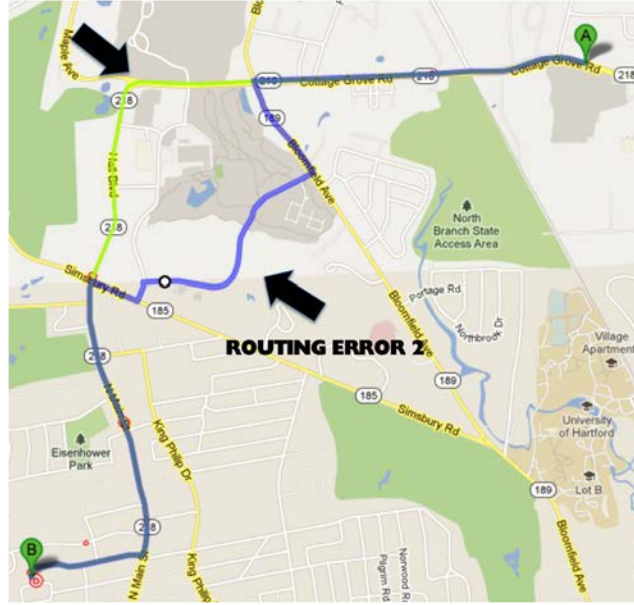


Figure 8: The two types of routing errors are shown on a map. The green line represents the ground truth, and the purple line represents the predicted route

These errors show that the initial predictions for trips made needs room for improvement. However, immediate and significant improvement can be made with some user input. As described in 5.4.4, users checked and corrected their trips made, which were cached as “hotroutes”. Any future trip made between the same origin and destination used the cached hotroute and corresponding transportation mode if the raw location points matched up with the stored route. On average, a total of 49 hotroutes per user were stored, with an average of 1.56 hotroutes per origin-destination pair. Considering these values, it is plausible to develop interfaces to prompt users of an automated travel diary system to check their trips made without creating a large burden for the user. Future work will investigate a system with more methods for user input to increase the accuracy of the trip determination algorithm.

5.4.6 Errors in Trip start time and end time

During the manual correction phase, the testers did not correct the trip start or end time, due to the amount of work required to accurately log the start and end time for trips made. However, the error range in the start and end time of trips can be calculated based on the accelerometer and Wifi readings during the trips made.

In the ideal scenario, movement is detected by the accelerometer, which leads to a scanning of WiFi beacons to determine if a significant location change has occurred. It is then known that a trip has started within the 2 minute window in which these

user	Route Match	Route Error 1	Route Error 2
1	45.3	41.4	29.8
2	40.5	44.9	34.6
3	59.3	30.8	11.8
4	85.9	5.5	11.6
5	94.1	4.2	4.2
6	90.1	9.5	5.7
user	Route Match	Route Error 1	Route Error 2
1	79.3	9.2	13.3
2	83.5	6.2	13.3
3	87.8	8.9	7.0
4	90.6	9.1	5.7
5	95.4	3.0	3.6
6	93.4	5.0	4.0

Table 2: Routing Errors. The table above represents the error metrics when no user corrections are used for six different users. The table below represents the error metrics when user corrections are used.

Predicted\True	Bike	Drive	Transit	Walk
Bike	66.1	30.8	1.8	1.3
Drive	19.2	53.1	19.2	8.5
Transit	5.0	9.4	85.0	0.6
Walk	2.5	8.6	0.5	88.4
Predicted\True	Bike	Drive	Transit	Walk
Bike	81.9	15.2	0.6	2.3
Drive	10.4	80.4	9.7	0.5
Transit	4.0	5.8	90.1	0.1
Walk	2.5	8.0	0.5	89.0

Table 3: Transportation Mode Errors. The table above represents the error metrics when no user corrections are used for six different users. the table below represents the error metrics when user corrections are used.

sensor events occurred. Of the 1850 trips evaluated, 48% of trips fell into this category.

In some cases it takes two to three cycles for the Wifi scan to identify a significant location change. The reason for this is if the intersection of the set of beacons seen at subsequent WiFi access point scans is null, then it is assumed that movement occurred during the time between the two scans. However, the range of WiFi access points vary in different environments, anywhere from 50m for 802.11b routers to 200m for 802.11g routers. A person can walk to start a trip and still be in range of an access point in 2 minutes. Of the 1850 trips evaluated, 36% of trips were detected within 4-6 minutes of their true start time.

In the last 16% of cases, a variety of scenarios lead to the system being able to pinpoint the start time of the trip to under 6 minutes. This includes a variety of reasons described in Section 5.1, such as user behaviors such as a phone turned off, phone behaviors such as the sensors unable to obtain readings, or the nature of the trip, which are often short trips of under 1 kilometer. In these scenarios, the algorithm described in Section 4.6 is used to determine the start/end time of trips.

Calculating the end times of a trip is a much simpler process. While a person is moving during a trip, the duty cycles for location acquisition are much smaller, and thus can be used to determine a transition to the end of a trip. 40% of trip end times are accurate within 1 min. 44% of trip end times are accurate within 2 minutes. The reason for the increase is the sampling period of the GPS sensor is increased in certain scenarios, such as when a person is traveling greater than a particular speed, or traveling along a hotroute. The last 16% of trips the system is unable to detect the end time within 2 minutes as the problems described above in detecting the start time of a trip cascade into the end time prediction problem.

5.4.7 Falsely detected trips and missed trips

Due to the issues described in Section 5.1, it is often the case that trips are falsely detected, or trips are completely missed. A total of 33 trips were not identified by the system (missed trips), and 39 were identified as trips when no movement occurred (false trips). The 33 missed trips occurred were all 2 kilometers or less. Falsely detected trips occur as a result of inaccurate location points which slip through the filter described in Section 4.1. Certain sequences of inaccurate location points make it appear that a trip is occurring, which fools the system into generating trips. Missed trips are a result of two problems. As described in Sections 4.6 and 5.4.6, not all movements are captured, leading to a situation where location points exist only at the origin and destination of trips. In situations where the points at the destination have a high horizontal accuracy, it is difficult to determine if the point is a faulty reading from WiFi/cell tower positioning, or if the person actually made the trip. Thus, there are thresholds set, by horizontal accuracy and number of points seen, for which any location points under the threshold will set a valid destination. Increasing this threshold will increase the number of faulty trips made, therefore, there is a balance between missing, and falsely identifying trips.

6 CONCLUSION AND FUTURE WORK

The research presented in this paper represents a first step in introducing a complete, end-to-end, and battery efficient automated travel diary system which operates on smartphones, and describing the number of difficulties of using smartphones rather than GPS loggers, as the main data collection tool. Smartphones are ubiquitous and are equipped with the technology needed to be used as a data collection device in travel diary systems, but there are a number of problems (from user behaviors, and inaccuracies/inconsistencies in smartphone sensor data) that must be overcome before they can be used in a wide-scale automated travel diary system. In this paper, large scale tests have allowed us to catalogue a large list of problems with smartphones and solutions found from real devices in the field for over 9 months. These early results from the system are promising, and more work has to be done evaluating the trip metrics defined in this paper and analyzing how small amounts of human input can increase the accuracy of these metrics. Identifying the start and end locations of trips have shown to be very accurate, while improvements are needed to more accurately identify the start/end time of trips, route, and mode taken. First, the research showed that the start/end time of 16% of trips could not be identified within a 6 minute window. The problem here lies with the mobile application, whose parameters for detecting motion need further experimentation. Accurately identifying the route taken is a larger research problem. With sparse location data, the amount of improvement on the current algorithm is limited. However, fusing accelerometer, and magnetometer data with location data can improve on accurately identifying routes by better identifying turns and changes in direction. The mode determination portion of the algorithm can also be improved by adding features which come from the accelerometer and magnetometer. However, all these improvements come at a cost of reducing the battery life of phones - and the tradeoff between improving the metrics for trip determination and increasing battery usage is a tricky balancing act. In future research, any trip determination system should explicitly declare the battery consumption compared with the metrics used in this paper: these are the basic measures to be used to determine the quality of an automated trip diary system.

Another area for improvement is expanding upon the six person test to obtain more data for the accuracy of the trip determination algorithm. Since, the three month long data collection and user input required for the analysis of the trip determination algorithm is quite intensive, the test was limited to a small group of people. Validating the ground truth for all the trips during this extended period of time was an arduous task, but can be made easier by building better web tools to help the researcher and users to make this process easier to expand the number of users who can validate trips.

Aside from these goals for future research, there are a few research directions which were not tackled in this automated travel diary system. Travel diaries always record contextual information about a trip, for example, the number of passengers in a car, or the purpose of a trip. There is work to be done to derive these pieces of data from the smartphone, either actively by prompting users, passive detection, or a combination of both. For this system to be of use to researchers building travel

behavior models, trip attributes like these must be annotated. An area to explore is the use of third party APIs such as Yelp, Foursquare, or Google Places to obtain trip purpose information and design a system which prompts the user for feedback on their trip purpose.

ACKNOWLEDGEMENTS

The authors would like to acknowledge Thejo Kote, David Palmer, Ram Jayaraman for their hard work and the University of California Transportation Center for funding this research.

References

- [1] E. D., H. Timmermans, and L. V. Veghel, “Effects of data collection methods in travel and activity research,” Prepared for European Institute of Retailing and Services Studies, Tech. Rep., 1996.
- [2] P. R. Stopher, K. Kockelman, S. P. Greaves, and E. Clifford, “Reducing burden and sample sizes in multiday household travel surveys,” *Transportation Research Record*, vol. 2064, pp. 12–18, 2008.
- [3] J. Wolf, R. Guensler, and W. Bachman, “Elimination of the travel diary: An experiment to derive trip purpose from gps travel data.” in *Transportation Research Record No. 1768*, *Transportation Research Board*, Washington, DC, USA, 2001, pp. 125–134.
- [4] P. R. Stopher, N. Swann, and C. FitzGerald, “Using an odometer and a gps panel to evaluate travel behaviour change programs,” in *11th TRB National Planning Applications Conference*, Daytona Beach, FL, USA, May 2007.
- [5] L. M. T. M. . A. C. Wolf, J., *Transport survey quality and innovation*. Oxford: Pergamon Press, 2003, ch. Trip rate analysis in GPS-enhanced personal travel surveys., pp. 483–498.
- [6] J. Wolf, *Travel survey methods: Quality and future directions*. Oxford: Elsevier, 2006, ch. Applications of new technologies in travel surveys, pp. 531–544.
- [7] S. Bricka, “Non-response challenges in gps-based surveys,” in *8th International Conference on Travel Survey Methods*, Annecy, France, May 2008.
- [8] D. N. P. P. i. t. U. Nielsen, Smartphones Account for Half of all Mobile Phones, “<http://www.nielsen.com/us/en.html>,” Tech. Rep., 2012.
- [9] Y. Chon, E. Talipov, H. Shin, and H. Cha, “Mobility prediction-based smartphone energy optimization for everyday location monitoring,” in

- Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '11. New York, NY, USA: ACM, 2011, pp. 82–95. [Online]. Available: <http://doi.acm.org/10.1145/2070942.2070952>
- [10] D. Ferreira, A. Dey, and V. Kostakos, “Understanding human-smartphone concerns: A study of battery life,” in *Pervasive Computing*, ser. Lecture Notes in Computer Science, K. Lyons, J. Hightower, and E. Huang, Eds. Springer Berlin / Heidelberg, 2011, vol. 6696, pp. 19–33.
 - [11] J. Wolf, “Applications of new technologies in travel surveys,” in *7th International Conference on Travel Survey Methods, Costa Rica*, 2004.
 - [12] P. R. Stopher and S. P. Greaves, “Household travel surveys: Where are we going?” *Transportation Research Part A: Policy and Practice*, vol. 41, no. 5, pp. 367–381, 2007, bridging Research and Practice: A Synthesis of Best Practices in Travel Demand Modeling. [Online]. Available: <http://www.sciencedirect.com/science/article/B6VG7-4M6SBB1-2/2/65f88300cd48e4189ca4bd661243ba5d>
 - [13] D. P. Wagner, E. Murakami, and M. Guindon, “Using gps for measuring household travel in private vehicles,” in *6th TRB Conference on the Application of Transportation Planning Methods*, Dearborn, Michigan, USA, 1997.
 - [14] L. a. t. d. c. t. g. p. s. f. p. t. s. F. R. t. F. F. H. A. Battelle Transportation Division, “<http://fhwa.dot.gov/ohim/trb/reports.htm>,” Tech. Rep., 1997.
 - [15] E. Murakami and D. P. Wagner, “Can using global positioning system (gps) improve trip reporting?” *Transportation Research C*, pp. 7, 149–165, 1999.
 - [16] P. Stopher, C. FitzGerald, and M. Xu, “Assessing the accuracy of the sydney household travel survey with gps,” *Transportation*, vol. 34, pp. 723–741, 2007, 10.1007/s11116-007-9126-8. [Online]. Available: <http://dx.doi.org/10.1007/s11116-007-9126-8>
 - [17] J. Zmud and J. Wolf, “Identifying the correlates of trip misreporting—results from the california statewide household travel survey gps study,” in *10th IATBR*, Lucerne, 2003.
 - [18] G. Draijer, N. Kalfs, and J. Perdok, “Global positioning system as data collection method for travel research,” *Transportation Research Record*, vol. 1719, pp. 147–153, 2000.
 - [19] J. Wolf, “Using gps data loggers to replace travel diaries in the collection of travel data,” Ph.D. dissertation, Georgia Institute of Technology, School of Civil and Environmental Engineering, Atlanta, Georgia, USA, 2000.
 - [20] R. de Jong and W. Mensonides, “Wearable gps device as a data collection method for travel research, its working paper 03-02,” Institute of Transport Studies, Sydney, Tech. Rep., 2003.

- [21] J. Wolf, S. Schöf, U. Samaga, and K. Axhausen, "80 weeks of gps-traces: Approaches to enriching trip information," in *83rd Annual Meeting of the Transportation Research Board*, Washington, D.C., USA, 2004.
- [22] E. Chung and A. Shalaby, "Development of a trip reconstruction tool to identify traveled links and used modes for gps-based personal travel surveys," in *83rd Annual Meeting of the Transportation Research Board*, Washington, D.C., USA, January 2004.
- [23] E. Chung, T. Ye, and A. Shalaby, "An integrated gps-gis system for personal travel surveys," in *GEOIDE Annual Conference*, Quebec, May 2004.
- [24] S. Greaves, "A panel approach to evaluating voluntary travel behavior change programs—south australia pilot survey," in *Transportation Research Board 85th Annual Meeting*, 2006.
- [25] S. T. Doherty, N. Noel, M.-L. Gosselin, C. Sirois, and M. Ueno, "Moving beyond observed outcomes: integrating global positioning systems and interactive computer-based travel behavior surveys." *Transportation Research Circular*, pp. 449–466, 2001.
- [26] E. Hato, T. Mitani, and S. Itsubo, "Development of moals (mobile activity loggers supported by gps-phones) for travel behavior analysis," in *Transportation Research Board 85th Annual Meeting*, 2006.
- [27] C. Williams, J. Auld, A. K. Mohammadian, and P. C. Nelson, "An automated gps-based prompted recall survey with learning algorithms," *Transportation Letters: The International Journal of Transportation Research*, vol. 1, pp. 59–79, 2009.
- [28] P. R. Stopher, P. Bullock, and F. Horst, "Exploring the use of passive gps devices to measure travel," in *Proceedings of the 7th International Conference on Applications of Advanced Technologies to Transportation*, S. N. K. C. P. Wang, S. Medanat and G. Spring, Eds. Reston, VA: ASCE, 2002, pp. 959–967.
- [29] C. A. Stopher, P. and P. Bullock, "Gps surveys and the internet," in *27th Australasian Transport Research Forum*, Adelaide, September 2004.
- [30] M. E. H. Lee-Gosselin, S. T. Doherty, and D. Papinski, "Internet-based prompted recall diary with automated gps activity-trip detection: System design," in *TRB 85th Annual Meeting Compendium of Papers CD-ROM*, Transportation Research Board, Washington, D.C., USA, 2006.
- [31] S. Y. A. Tsui and A. S. Shalaby, "Enhanced system for link and mode identification for personal travel surveys based on global positioning systems. transportation research record, 1972," in *Transportation Research Board*, Washington, D. C., USA, 2006, pp. 38–45.

- [32] P. Bachu, R. Dudala, and S. Kothuri, "Prompted recall in global positioning survey: Proof of concept study," *Transportation Research Record*, vol. 1768, pp. 106–113, 2001.
- [33] M. Wermuth, C. Sommer, and M. Kreitz, *Impact of new technologies in travel surveys*. Pergamon, Oxford, 2003, ch. Transport Survey Quality and Innovation, pp. 151–180.
- [34] S. Krygsman, J. Nel, and T. de Jong, "The use of cellphone technology in activity and travel data collection in developing countries." in *18th International Conference on Transport Survey Methods*, Annecy, France, 2008.
- [35] P. R. Stopher, *Transport survey methods: Keeping up with a changing world*. Bingley, UK: Emerald, 2009, ch. Collecting and processing data from mobile technologies.
- [36] Y. Asakura, E. Hato, Y. Nishibe, T. Daito, J. Tanabe, and H. Koshima, "Monitoring travel behaviour using pbs based location positioning service system," in *6th ITS World Congress*, Toronto, Canada, 1999.
- [37] Y. Asakura and E. Hato, "Tracking survey for individual travel behaviour using mobile communication instruments," *Transportation Research C*, vol. 12(3/4), pp. 273–291, 2004.
- [38] K. Sugino, S. Yano, E. Hato, and Y. Asakura, "Empirical analysis of sightseeing behaviour using probe person survey data," *Proceedings of Infrastructure Planning*, vol. 32, 2005.
- [39] S. Itsubo and E. Hato, "Effectiveness of household travel survey using gps-equipped cell phones and web diary: comparative study with paper-based travel survey," in *Transportation Research Board 85th Annual Meeting*, 2006.
- [40] H. Yatsumoto, T. Kitazawa, S. Nakagawa, A. Okamoto, and Y. Asakura, "Analysis of route choice behavior under flexible toll system of urban expressway based on probe person trip survey," in *33rd Meeting of Infrastructure Planning*, 2006.
- [41] T. Mitani, "Study on the applicability of travel information provision system supported by probe person survey," Ph.D. dissertation, Ehime University, 2005.
- [42] S. Reddy, M. Mun, J. A. Burke, D. Estrin, M. Hansen, and M. B. Srinivastava, "Using mobile phones to determine transportation modes," 2008.
- [43] L. Liao, D. J. Patterson, D. Fox, and H. Kautz, "Learning and inferring transportation routines," *Artificial Intelligence*, vol. 171, no. 5-6, pp. 311 – 331, 2007. [Online]. Available: <http://www.sciencedirect.com/science/article/B6TYF-4N49VP9-1/2/8f05b8caf7327ceb8762ab5e1b95efc9>

- [44] Y. Zheng, Y. Chen, Q. Li, X. Xie, and W.-Y. Ma, “Understanding transportation modes based on gps data for web applications,” *ACM Trans. Web*, vol. 4, pp. 1:1–1:36, January 2010. [Online]. Available: <http://doi.acm.org/10.1145/1658373.1658374>
- [45] P. Gonzalez., J. Weinstein, S. Barbeau, M. Labrador, P. Winters, N. Georggi, and R. Perez, “Automating mode detection for travel behaviour analysis by using global positioning systems-enabled mobile phones and neural networks,” *IET Intelligent Transport Systems*, vol. Vol. 4, Iss. 1, pp. 37–49, 2010.
- [46] L. Stenneth, O. Wolfson, P. S. Yu, and B. Xu, “Transportation mode detection using mobile phones and gis information,” in *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ser. GIS ’11. New York, NY, USA: ACM, 2011, pp. 54–63. [Online]. Available: <http://doi.acm.org/10.1145/2093973.2093982>
- [47] X. Y., D. Low, T. Bandara, P. Pathak, H. B. Lim, D. Goyal, J. Santos, C. Cottrill, F. Pereira, C. Zegras, and M. Ben-Akiva, “Transportation activity analysis using smartphones,” Massachusetts Institute of Technology, Cambridge, MA, USA, Tech. Rep., 2011.
- [48] A. Thiagarajan, L. Ravindranath, H. Balakrishnan, S. Madden, and L. Girod, “Accurate, low-energy trajectory mapping for mobile devices,” in *Proceedings of the 8th USENIX conference on Networked systems design and implementation*, ser. NSDI’11. Berkeley, CA, USA: USENIX Association, 2011, pp. 20–20. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1972457.1972485>
- [49] P. Stopher and N. Speisser, “Evaluation of gps device properties for a possible use in future household travel surveys,” The University of Sydney, Sydney, Australia, Tech. Rep. Working Paper ITLS-WP-11-08, April 2011.
- [50] K. N. Truong, J. A. Kientz, T. Sohn, A. Rosenzweig, A. Fonville, and T. Smith, “The design and evaluation of a task-centered battery interface,” in *Proceedings of the 12th ACM international conference on Ubiquitous computing*, ser. Ubicomp ’10. New York, NY, USA: ACM, 2010, pp. 341–350. [Online]. Available: <http://doi.acm.org/10.1145/1864349.1864400>
- [51] A. Rahmati, A. Qian, and L. Zhong, “Understanding human-battery interaction on mobile phones,” in *Proceedings of the 9th international conference on Human computer interaction with mobile devices and services*, ser. MobileHCI ’07. New York, NY, USA: ACM, 2007, pp. 265–272. [Online]. Available: <http://doi.acm.org/10.1145/1377999.1378017>
- [52] E. Oliver, “The challenges in large-scale smartphone user studies,” in *Proceedings of the 2nd ACM International Workshop on Hot Topics in Planet-scale Measurement*, ser. HotPlanet ’10. New York, NY, USA: ACM, 2010, pp. 5:1–5:5. [Online]. Available: <http://doi.acm.org/10.1145/1834616.1834623>

- [53] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, “Limits of predictability in human mobility,” *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010. [Online]. Available: <http://www.sciencemag.org/content/327/5968/1018.abstract>
- [54] L. Song, D. Kotz, R. Jain, and X. He, “Evaluating next-cell predictors with extensive wi-fi mobility data,” *IEEE Trans. Mobile Comput.*, vol. 5(12), pp. 1633–1649, December 2006.
- [55] F. Alizadeh-Shabdiz and E. J. Morgan, “System and method for estimating positioning error within a wlan-based positioning system,” USA Patent 2008/0 108 371 A1, 2008.
- [56] 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, and B. Schilit, “Place lab: Device positioning using radio beacons in the wild,” in *Pervasive Computing*, ser. Lecture Notes in Computer Science, H. Gellersen, R. Want, and A. Schmidt, Eds. Springer Berlin / Heidelberg, 2005, vol. 3468, pp. 301–306.
- [57] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell, “The jigsaw continuous sensing engine for mobile phone applications,” in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys ’10. New York, NY, USA: ACM, 2010, pp. 71–84. [Online]. Available: <http://doi.acm.org/10.1145/1869983.1869992>
- [58] Z. Zhuang, K.-H. Kim, and J. P. Singh, “Improving energy efficiency of location sensing on smartphones,” in *Proceedings of the 8th international conference on Mobile systems, applications, and services*, ser. MobiSys ’10. New York, NY, USA: ACM, 2010, pp. 315–330. [Online]. Available: <http://doi.acm.org/10.1145/1814433.1814464>
- [59] J. Paek, J. Kim, and R. Govindan, “Energy-efficient rate-adaptive gps-based positioning for smartphones,” in *Proceedings of the 8th international conference on Mobile systems, applications, and services*, ser. MobiSys ’10. New York, NY, USA: ACM, 2010, pp. 299–314. [Online]. Available: <http://doi.acm.org/10.1145/1814433.1814463>
- [60] D. H. Kim, Y. Kim, D. Estrin, and M. B. Srivastava, “Sensloc: sensing everyday places and paths using less energy,” in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys ’10. New York, NY, USA: ACM, 2010, pp. 43–56. [Online]. Available: <http://doi.acm.org/10.1145/1869983.1869989>
- [61] Y. Ma, R. Hankins, and D. Racz, “iloc: a framework for incremental location-state acquisition and prediction based on mobile sensors,” in *Proceeding of the 18th ACM conference on Information and knowledge management*, ser. CIKM ’09. New York, NY, USA: ACM, 2009, pp. 1367–1376. [Online]. Available: <http://doi.acm.org/10.1145/1645953.1646126>

- [62] I. Constandache, S. Gaonkar, M. Sayler, R. Choudhury, and L. Cox, “Enloc: Energy-efficient localization for mobile phones,” in *INFOCOM 2009, IEEE*, 2009, pp. 2716–2720.
- [63] K. Lin, A. Kansal, D. LyMBERopoulos, and F. Zhao, “Energy-accuracy trade-off for continuous mobile device location,” in *Proceedings of the 8th international conference on Mobile systems, applications, and services*, ser. MobiSys ’10. New York, NY, USA: ACM, 2010, pp. 285–298. [Online]. Available: <http://doi.acm.org/10.1145/1814433.1814462>
- [64] A. K. Jain, M. N. Murty, and P. J. Flynn, “Data clustering: a review,” *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, Sep. 1999. [Online]. Available: <http://doi.acm.org/10.1145/331499.331504>
- [65] “Google directions api, <http://code.google.com/apis/maps/documentation/directions/>,” Tech. Rep., 2012.
- [66] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, pp. 5–32, 2001, 10.1023/A:1010933404324. [Online]. Available: <http://dx.doi.org/10.1023/A:1010933404324>
- [67] GTFS-Data-Exchange, “<http://www.gtfs-data-exchange.com/>,” Tech. Rep.
- [68] J. Bentley and T. Ottmann, “Algorithms for reporting and counting geometric intersections,” *IEEE Transactions on Computers*, vol. 28, pp. 643–647, 1979.