# Probabilistic Inference of Unknown Locations

## Exploiting Collective Behavior when Individual Data is Scarce

Joshua Blumenstock*
University of Washington
Information School
Seattle, WA
joshblum@uw.edu

Ramkumar Chokkalingam
University of Washington
Information School
Seattle, WA
ramar@uw.edu

Vijay Gaikwad
University of Washington
Information School
Seattle, WA
vijaygkd@uw.edu

Sashwat Kondepudi
University of Washington
Information School
Seattle, WA
sk08@uw.edu

## ABSTRACT

In recent years there has been a proliferation in the use of large-scale, passively collected digital trace data to study the mobility and migration patterns of individuals in developing countries. Analysis of mobile phone and social media data, among other sources, has immediate policy applications that range from disease monitoring and city planning to disaster management and humanitarian relief. Unfortunately, existing methods for mining location-based information from passively collected data are generally not well suited to a large number of individuals in developing countries. This is in part due to the fact that technology use is quite heterogeneous, and that the lower intensity use patterns of many individuals produces a sparser digital trace.

In this paper, we present a method for predicting the approximate location of a mobile phone subscriber that is more appropriate to contexts where the signal generated by each individual may be intermittent, but the collective population generates a large amount of data. This method works well when, for instance, an individual is not consistently active on the network or when the phone is off. Our model uses a nonparametric approach to probabilistically interpolate locations, and has the advantage of associating a confidence with each prediction. We test this method on a large dataset of anonymized mobile phone records from Afghanistan, and find that we can correctly predict a subscriber's unknown location in 76%-95% of cases, and that on average our predicted location is off by 0.2-1.9 kilometers.

## Categories and Subject Descriptors

J.4 [**Computer Applications**]: Social and Behavioral Sciences; G.3 [**Mathematics of Computing**]: Probability and Statistics

*Correspondence should be addressed to joshblum@uw.edu


*ACM DEV-5 (2014),* December 5–6, 2014, San Jose, CA, USA.

http://dx.doi.org/10.1145/2674377.2674387.

## Keywords

Mobility; call detail records; probabilistic inference; sparsity; ICTD

## 1. INTRODUCTION AND MOTIVATION

The expansion of mobile phone networks and the rapid adoption of Information and Communication Technologies (ICTs) in developing countries has created a unique opportunity for researchers and policymakers to obtain large-scale, reliable, quantitative data on the behaviors of individuals who have historically been difficult to monitor. The increasing availability of such data raises important ethical considerations regarding privacy and population legibility [17, 18], but it also creates tremendous opportunity for researchers and policymakers interested in better understanding and developing appropriate policy for historically marginalized populations.

One particular area of promise is the use of spatiotemporal data, such as the traces generated through the use of mobile phones or other geolocated digital devices, for the modeling and measurement of human mobility. Such data generally have spatial and temporal markers that approximately locate the person using the device at the time when she takes an action, such as making a phone call, sending an SMS, posting a Tweet, or downloading emails or webpages. As such, these data can allow for the granular reconstruction of the trajectory taken by that individual through space over time, and this property of the data has generated a spate of recent research on human mobility [10, 19, 14].

Such research has many applications relevant to policy in developing countries. Improved models of mobility based on mobile phone data can better enable epidemiologists and public health officials to understand and interrupt the spread of malaria and other diseases [20, 8]. More accurate data on the location and movement of populations can also improve disaster response and planning [2, 11]. Particularly in urban areas, location data from phones can improve urban layout and city planning [16]. More generally, internal and international migration rates are a key input to economic policy, and phone data may enable more fine-grained perspectives on the location and structure of local populations [5, 6, 3, 4].

In this paper, we present a method, model, and scalable algorithm for continuously inferring the location of an individual at an arbitrary point in time, based on a limited and potentially quite sparse set of spatiotemporal observations. We develop and calibrate this method using a large dataset of anonymized mobile phone

records from Afghanistan, but in principle the method could be applied to any dataset where the location of individuals are observed intermittently over time. In the mobile phone context, the use case we have in mind is one in which an individual might only make a handful of calls on a given day or in a given week; the method in this paper allows for probabilistic inference of the individual's location at points in times when the individual is not active on the network and the actual location is unknown.

Specifically, we develop a technique that relies on past trajectories taken by the individual of interest, as well as the trajectories taken by similar individuals in the past. Our approach constructs a probability distribution of likely locations for each person at each point in time, where the probability of each location is determined by the universe of observed trajectories. Trajectories between similar users and trajectories with similar temporal features (such as the time of day or day of week) contribute a greater weight to the probability distribution. A primary advantage of the method we present is that the weights need not be specified ex ante; rather, the model is non-parametric and the relevant parameters can be efficiently learned via cross-validation. We also present two scalable implementations of the algorithm, one in a Map-Reduce paradigm that is optimized for deployment on a cluster, as well as a memory-intensive version that is better suited to a single node with a large amount of memory.

In the following section, we describe related work on similar problems and discuss the primary ways in which this method is different from previous research. Section 3 then describes the model, beginning with an intuitive overview and proceeding to a formal description of the model and algorithm. Section 4 describes the data on which we test the model, and Section 5 presents the empirical results with a discussion of simple extensions that can significantly improve performance. Section 6 concludes with a discussion of the advantages and limitations of our approach, and outlines next steps for future work.

## 2. RELATED WORK

A rich literature in transport planning and civil engineering is concerned with modeling and predicting human movement [1]. More recently, the increasing availability of fine-grained data from mobile phone networks and other GPS-enabled devices has created new opportunities for building high-resolution models of human movement [16]. While a great deal of the focus of this literature has been on predicting aggregate population flows [5], a handful of recent studies have utilized mobile phone call detail records to predict individual locations and trajectories.

The focus of most of these efforts have been on using machine learning techniques including decision trees [15], markov models [21, 12], and hybrid approaches [21] to predict the next location of an individual given a sequence of previous locations. Many of these studies additionally incorporate rich GPS data beyond the simple metadata from call and SMS transactions. Both [9] and [12] take the basic algorithm a step further and develop frameworks for deploying a destination prediction system.

In the studies most closely related to our own, [7] and [13] demonstrate the value of incorporating collective travel patterns for predicting individual locations. In both cases, the authors use data from the Boston metropolitan area to infer future movements based on past patterns of travel. The predictive model they develop additionally incorporates known geographic features of the physical environment, which they believe is a proxy for the activities that affect travel decisions. After careful calibration on a sample of highly active mobile phone users, the authors report being able to predict locations with an accuracy of roughly 1.5 kilometers. In a related extension, [21] integrate a markov-based predictor to predict the movements of 106 individuals associated with MIT, and find that the hybrid predictor performs significantly better than a simpler model.

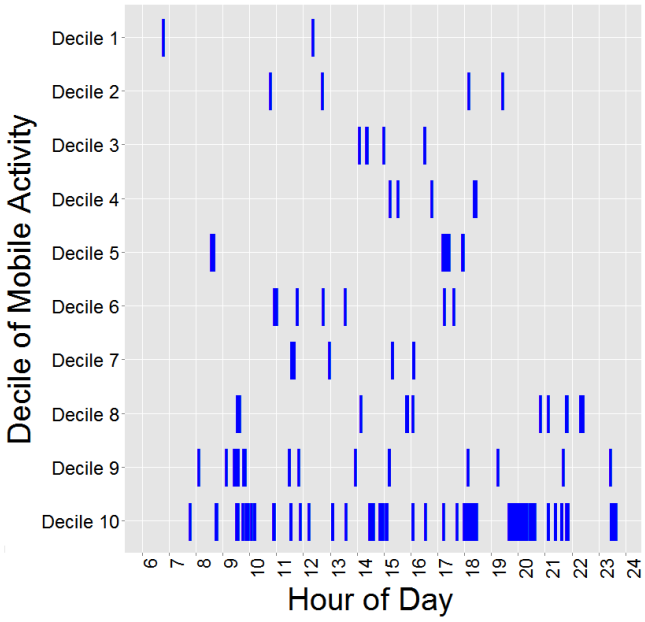


Figure 1: **Sparsity of data.** The figure shows all of the observed transactions (blue vertical lines) for ten different subscribers in Afghanistan. Subscribers are sampled randomly, one from each decile of mobile activity, wherein the median subscriber (Decile 5) makes 6 unique transactions per day.

Our work builds upon and distinguishes itself from previous research in several critical aspects. First and most importantly, it is designed to allow for inference in the developing country context, where the data is generated by a deeply heterogeneous population of subscribers, the majority of whom do not use their devices at regular and consistent intervals. The model we develop provide predictions and confidence intervals for those predictions, even when data is sparse. The confidence will be higher for regular network users, but gracefully scales to individuals with sporadic activity.

This issue of data sparsity can be seen in Figure 1, which shows the number of transactions observed in a typical day for subscriber in Afghanistan. To construct Figure 1, all subscribers are ranked by the average number of transactions per day, and then one subscriber is sampled randomly from each decile of activity. While the representative subscriber in the top decile is very active, and consistently uses his or her phone from roughly 7:30am until 11:30pm, below the 80th percentile usage is actually quite sparse, and for many subscribers there are long periods of time when no transactions are made and thus the subscriber's location is not directly observed. These statistics are in contrast to the typical dataset used by much of the literature discussed above, where, for instance [7] restrict their analysis to subscribers with "100 network connections per day (with individual inter-event time below 1 hour in 75 percent of the cases)", and [21] have tens of thousands of observations per individual. As we show later in Section 5.3, such a restriction would eliminate over 99 percent of the subscribers from our population. The performance would certainly improve on such a subsample, but this restriction would severely limit the relevance of the method to developing countries.

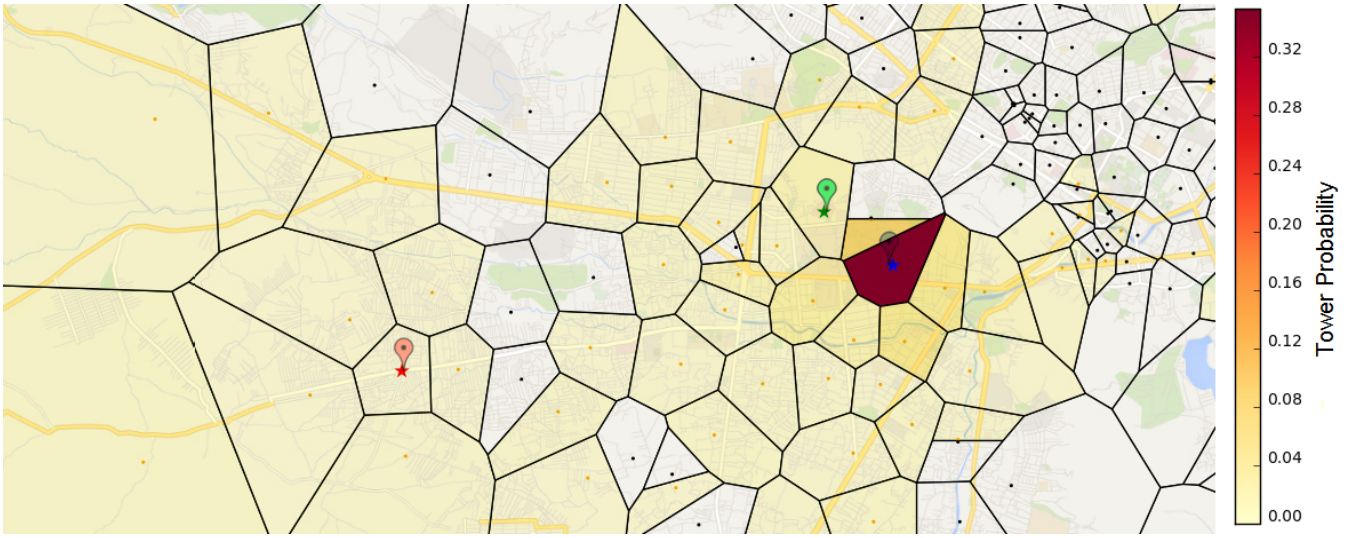


Figure 2: **Known and predicted locations of a single individual**. The green star indicates the start location and the red star indicates the end location, both of which are directly observed. The voronoi cells are colored according to the probability that the individual is in those locations, as determined by the inference algorithm. The actual location (held back for the purposes of prediction) is indicated by the blue star. Predicted locations tend to lie along major roadways, and the top predicted location lies along the fastest road route between the start and end location, though off of the shortest path between the two locations.

Our approach is further designed to function well in information-poor environments. Thus, we explicitly avoid incorporating any structured input data such as the location or type of geographical features such as roads and other points of interest. While such features would likely constrain trajectories in a manner that would improve predictive accuracy, in many of the environments for which our method is designed, structured data of this nature does not exist. Similarly, the approach we describe is intentionally non-parametric, and relies as little as possible on *a priori* knowledge such as the relative weights to be placed on trajectories of the same individual vs. collective behavior. Instead, we develop a generalized framework that relies upon the notion of "trajectory similarity" to determine the weight and confidence associated with each predicted location. As described in Section 3.2.2, the shape of the similarity function, as well as any associated parameters, can be learned via cross-validation.

# 3. METHODS AND MODEL

## 3.1 Overview of approach

The model we develop enables the probabilistic interpolation of an individual's location when the true location is unknown. Intuitively, given two observations of the individual's spatiotemporal location (the start time and location and the end time and location), as well as a "query time" where the actual location of that individual is unknown, the algorithm constructs a probability distribution over possible locations for the unknown time. The probability distribution is built by analyzing the past trajectories of all other individuals for whom data is available, leveraging the behavioral patterns of the collective to make inferences about the individual. Similar trajectories – where similarity is carefully defined in the following section – are weighted more heavily than dissimilar trajectories.

A graphical representation of the output of the algorithm is shown in Figure 2. We construct a sample query for a given subscriber, which consists of the metadata from two mobile phone transactions. The first transaction was routed through the cell phone tower marked with a green star, and the second transaction was routed through the cell phone tower marked with a red star. All of the other cell phone towers in the region are illustrated with yellow and black dots. The black lines create a voronoi division of the space, which we overlay on a Google Map of the geographic region. In this particular instance, the first transaction occurred at roughly 11:00 am and the second transaction occurred roughly an hour later on the same day. We then pick an intermediate query time in between the two transactions, and use our algorithm to construct a probability distribution over space at that query time. This probability distribution is used to color the voronoi cells, where the color indicates the predicted probability that the individual is in a given cell. In this figure, we chose a query time where the true location was known (but not included as training data for the algorithm); this location is indicated by a blue star.

In the instance depicted in Figure 2, note that the intermediate location – both the predicted location (indicated by the red cell) and the actual location (indicated by the blue star) lie off the direct pathway between the start (green star) and end (red star) locations. This illustrates one advantage of our method: rather than simple linear interpolation, which would predict a location along the direct path, our algorithm is able to recover the more common route taken between these locations, even without explicit knowledge of the road network or other physical landmarks. Here, it so happens that the direct path is blocked by two large buildings (Kabul Medical University and the Ministry of Higher Education), which forces the individual to take an arterial road to the southeast.

Aside from the primary prediction, indicated by the red cell, the algorithm also reveals useful information about the other lower-probability predicted locations. In particular, there are a large number of locations to which the model assigns low but non-zero probability, indicated by the pale yellow cells in the figure. Some of these locations are rather distant from the known start and end points, but generally these fall along major highways or transit routes.

## 3.2 Predictive Model

Formally, we are interested in predicting the location $l$ of an individual $u$ at time $t$, which we define as $Loc(u,t)$. We denote by $u^*$ a specific "query" individual with unknown location at a specific time $t^*$, but whose prior location $l_s^*$ was known at a time $t_s^*$ before $t^*$, and whose later location $l_f^*$ was known at a time $t_f^*$ after $t^*$. Our goal is thus to determine, for each possible location $l$, the probability that $u^*$ was at $l$ at time $t^*$,

$$P\left(Loc(u^*,t^*)=l \,\middle|\, \begin{array}{l} Loc(u^*,t_s^*)=l_s^* \\ Loc(u^*,t_f^*)=l_f^* \end{array}\right) = \frac{W_l\left(u^*,t^*\middle|t_s^*,t_f^*,l_s^*,l_f^*\right)}{\sum_i^L W_i\left(u^*,t^*\middle|t_s^*,t_f^*,l_s^*,l_f^*\right)} \tag{1}$$

where $W_l(\cdot)$ indicates the weight given to location $l$ at time $t^*$ for individual $u^*$, and is defined by

$$W_l(\cdot)=\sum_u^N \sum_{t_s\in e(u)} \sum_{\substack{t_f\in e(u)\\ \forall t_s<t_f}} \sum_{\substack{t\in e(u)\\ \forall t>t_s,\\ t<t_f}} S(u,u^*,t,t^*,t_s,t_s^*,t_f,t_f^*,l_s,l_s^*,l_f,l_f^*) \tag{2}$$

where $e(u)$ is the set of all timestamps observed for individual $u$ and $S(\cdot)$ indicates the similarity between a query event (denoted with asterisks) to the observed events (denoted without asterisks). We define this similarity as

$$S(\cdot)=h(u,u^*)\times k(t,t^*)\times k_s(t_s,t_s^*)\times k_f(t_f,t_f^*)\times g_s(l_s,l_s^*)\times g_f(l_f,l_f^*) \tag{3}$$

Thus, similarity is determined by four components: the similarity between two individuals $h(u,u^*)$; the similarity between the query time and observation time $k(t,t^*)$; the similarity in the start times and end times of the query and observation $k_{\{s,f\}}(t,t^*)$; and the similarity in the start and end locations of the query and observation $g(l,l^*)$. In principal, these similarity functions need not be paramaterized *a priori*, and could be learned in a supervised framework. For initial testing, however, we specify these components as follows:

1. $h(u,u^*)$: The similarity between two individuals $u$ and $u^*$ is a step function that assigns a different weight to past trajectories by the query individual

$$h(u,u^*)=\begin{cases}\alpha & \text{if } u=u^* \\ 1-\alpha & \text{otherwise}\end{cases}, \text{ where } \alpha\in[0,1]$$

2. $k(t,t^*)$: The similarity of a query time and the timestamp of a known event uses a simple kernel function that places higher weight on proximate events

$$k(t,t^*)=\frac{1}{|t-t^*|^\varepsilon}, \text{ where } \varepsilon >= 0$$

3. $k_{\{s,f\}}(t,t^*)$: The similarity in start and end times is defined by a step function that excludes events more than 30 minutes removed:

$$k_s(t_s,t_s^*)=k_f(t_f,t_f^*)=\begin{cases}1 & \text{if } |t_{\{s,f\}}-t_{\{s,f\}}^*|<30min \\ 0 & \text{otherwise}\end{cases}$$

4. $g_{\{s,f\}}(l,l^*)$: The similarity in locations is defined by a geometric distance function that places even weight on observations within a specified radius of the query location

$$g_s(l_s,l_s^*)=g_f(l_f,l_f^*)=\begin{cases}1 & \text{if } distance(l_{\{s,f\}},l_{\{s,f\}}^*) <= K \\ 0 & \text{otherwise}\end{cases}$$

### 3.2.1 Implementation and algorithm

We provide two scalable implementations of our model in Algorithm 1 and Algorithm 2. Algorithm 1 is a linear method best suited for a single-server, multi-threaded environment where the server has a large amount of memory. In Step 1 of the algorithm, the training data, which consists of all historical locations for all $N$ individuals, is redundantly indexed in memory. *LocationMap* indexes the training data by the user's location and time, while *UserMap* indexes by the anonymized user ID. After this indexing, which can be performed offline and re-indexed as new data is received, new queries can be executed on the fly and in parallel. To make the prediction for a query, first the start and end locations of the query user $u^*$ are found out for the query time $t^*$ (Step 2.1). In the next step all the users who satisfy the query conditions ($k_{\{s,f\}}$ and $g_{\{s,f\}}$) are found from the training data (Step 2.2). In the final step, predictions for the unknown location are made using the locations of the *MatchedUsers* from the training data using model (1) (Step 2.3).

Algorithm 2 utilizes a MapReduce framework and is optimized for a distributed environment with minimal requirements on the compute nodes. This is a batch processing approach where predictions are made on a sets of $M$ queries at a time. In Step 1, the algorithm finds the start and end locations and times of each of the $M$ queries. In Step 2, the mapping function emits records matching the query, where the start and end locations for each query are compared against each possible pair of records for every user in the training data. Thus, in a single pass of the data, all matching records are found for all $M$ queries. For efficiency, only the necessary records are emitted, reducing the number of relevant key-value pairs. Lastly, the reducing function makes predictions for the unknown locations of each query from the location records of the matched users.

The different implementations have distinct advantages. Algorithm 1 is appropriate in "real-time" environments where predictions need to be made on the fly in constant time, and where the model can be continuously updated with new training data. After initialization, the predicted locations can be computed in constant time without needing to re-train the model. However, this comes at the cost of an expensive initialization stage which stores the full dataset in memory, so the O($2N$) memory requirements may not scale to datasets with trillions of records. Thus, the time complexity of Algorithm 1 is O($N$) during the first pass and O($k$) afterwards, where $k$ is the number of matched records for a query. By contrast, Algorithm 2 is more effective when computational resources are scarce, or in distributed environments like Hadoop or Spark. However, Algorithm 2 is not practical in a streaming context since a new pass over the training data must be made for every batch of queries. While the memory requirements are O($N$), the queries execute in O($N+Mk$) for a batch of $M$ queries with $k$ matching records for every query.

### 3.2.2 Model fitting and cross-validation

As described above, our model has two parameters that must either be specified *a priori* or learned from the data: $\alpha$, which determines the relative weight placed on observations from the same individual vs. the collective, and ranges from 0 to 1. Low values of $\alpha$ emphasize collective behavior over previous trajectories of the individual, while high values heavily weight past individual trajec-

| Metric | Afghanistan | | | Sample | | |
|---|---|---|---|---|---|---|
| | Mean | SD | Median | Mean | SD | Median |
| Transactions per person | 276.67 | 735.66 | 109 | 332.41 | 963.91 | 127.0 |
| Unique locations observed | 8.96 | 11.2 | 5 | 9.2 | 12.52 | 5.0 |
| Radius of gyration | 23.59 | 52.21 | 4.27 | 24.30 | 52.81 | 4.801 |

Table 1: **Summary statistics of mobile phone record dataset.** Values are computed for a single month in 2011 from a dataset covering millions of individuals. We separately report statistics for the entire population of mobile phone subscribers, and for the random sample of 10,000 subscribers that are used in our experiments.

tories. Note, however, that for any given query there are generally many more collective matches than individual matches. Thus, as currently formulated, $\alpha$ should be viewed as the relative weight placed on any single match, rather than the total relative weight of the collective vs. the individual. We will return to this point later.

The second parameter, $\varepsilon$, determines the weight given to trajectories that have a known location closer to the query time of the query individual. Specifically, $\varepsilon$ represents the extent to which the weight of a matching query should be discounted by the difference between the query time and the time of the matching record. Given the kernel function $1/(t-t^*)^{\varepsilon}$, large values of $\varepsilon$ will decrease the weight of matches that are distant in time from the query.[1]

To find optimal values of $\alpha$ and $\varepsilon$, we adopt a supervised learning approach akin to leave-one-out cross-validation. We define our cost function in terms of the error in kilometers between the predicted and the actual location, as described in greater detail in Section 5.1 below. We then perform a grid search over a large range of values for both parameters, where we cross-validate on one month of data with a sample of 1,000 subscribers. In this dataset, we find optimal values at $\alpha = 0.99$ (emphasis on individual histories) and $\varepsilon = 4$ (moderate discounting of temporally distant matches). We will discuss our interpretation of these values in greater detail in section 6.

# 4. DATA AND CONTEXT

We test the probabilistic inference algorithm described above on a large dataset of anonymized mobile phone call detail records (CDR) from one of the largest mobile operators in Afghanistan. A CDR is generated any time a subscriber is involved in a transaction mediated by the network, such as a mobile phone call or a text message. Each record contains the anonymized identifiers of the individuals involved in the transaction, the timestamp of the event, as well as the identifier of the nearest mobile phone tower, which we can use to map each transaction to latitude and longitude coordinates. In total, this dataset contains the records of several billion mobile phone transactions initiated by several million unique individuals.

Basic summary statistics of the dataset are presented in Table 1 separately for the entire country and for a random subset of 10,000 subscribers that we use to test performance. In the course of a month, the median subscriber is observed slightly more than 100 times, and appears at 5 different physical locations. The distribution has a long right tail with a small number of high-volume subscribers driving the mean number of transactions to be several times larger than the median. By either metric, this is relatively sparse data, especially compared to the data used by most comparable techniques in the literature, which are often based on populations transacting more than 100 times *per day* [7, 21].

[1]In principle, additional components of the algorithm could be learned from the data, including the functional form of $h()$ or $k()$. We do not attempt that in this paper, but will return to this idea in Section 6.

# 5. RESULTS

## 5.1 Experimental framework

To evaluate the performance of our algorithm, we measure the extent to which it can accurately predict the held-out locations of individuals in the Afghanistan CDR dataset. We train the model following Section 3.2.1 using a complete month of call records for several million subscribers from 2011. We then draw a test sample of 10,000 subscribers randomly and without replacement from the list of all subscribers, and for each individual we randomly choose a single transaction that occurs at least one week after the month from which the training data was selected. This randomly selected transaction serves as the "query" transaction, whereby we pass the timestamp to the algorithm (along with the timestamp and location of the transaction before and after the query transaction) and then compare the probability distribution produced by the algorithm to the actual location.

We report performance in two ways. "Accuracy" indicates whether the predicted location is an exact match with the actual location. "Error Distance" gives the difference in kilometers between the predicted location and the actual location. The "Top Match" method simply selects the location that has the highest predicted probability given by our algorithm. "Top-*N* Location Match" generalizes this metrics to indicate whether the actual location is one of the three most probable locations. In the results described below, we compare the performance of our model with two simple baselines that are commonly used in the literature. The first baseline (Baseline 1) calculates the geographic midpoint between the start and end locations. For this baseline, we compute the error distance using the location of the actual midpoint; to compute accuracy we use the location of the tower nearest to the midpoint. The second baseline model (Baseline 2) predicts the most common (modal) location visited by the subscriber.

## 5.2 Performance benchmarks

Basic benchmarks of performance are reported in Table 2. When tested on the full set of 10,000 random individuals, the Top-1 predictor is 75.8% accurate at finding the true location. This represents a 7% (5 percentage point) improvement over the midpoint baseline and a 19% (12 percentage point) improvement over the modal tower baseline. The performance of the Top-3 Location Match predictor is more flexible and significantly better, resulting in 29% (20 percentage point) and 44% (28 percentage point) improvements over the two baselines, respectively. When accuracy is measured by the geographic distance between the true and predicted location, both the top-1 and top-3 algorithms similarly outperform both baselines. As can be seen in the right-most columns of Table 2, predictive accuracy is even better when tested on just the most active subscribers.

| Metric | Random Sample | | "Active" Subscribers | |
|---|---|---|---|---|
| | Average Accuracy | Error Distance | Average Accuracy | Error Distance |
| Top-3 Locations | 0.917 | – | 0.953 | – |
| Top-2 Locations | 0.881 | – | 0.935 | – |
| Top-1 Location | 0.758 | 1.878 | 0.877 | 0.220 |
| Baseline (midpoint location) | 0.711 | 1.908 | 0.836 | 0.193 |
| Baseline (modal tower) | 0.635 | 12.496 | 0.590 | 9.791 |

Table 2: **Model performance.** Accuracy of predictions and average error (in kilometers) for three versions of the inference algorithm and for the two baseline models described in section 5.1. Performance is reported separately for a random sample of subscribers and for a random population of active subscribers, where "active" is relative and is defined as individuals who make more than 80 calls per day. Error distance is not reported for the Top-2 and Top-3 methods, since the output is a list of locations rather than a single predicted location.

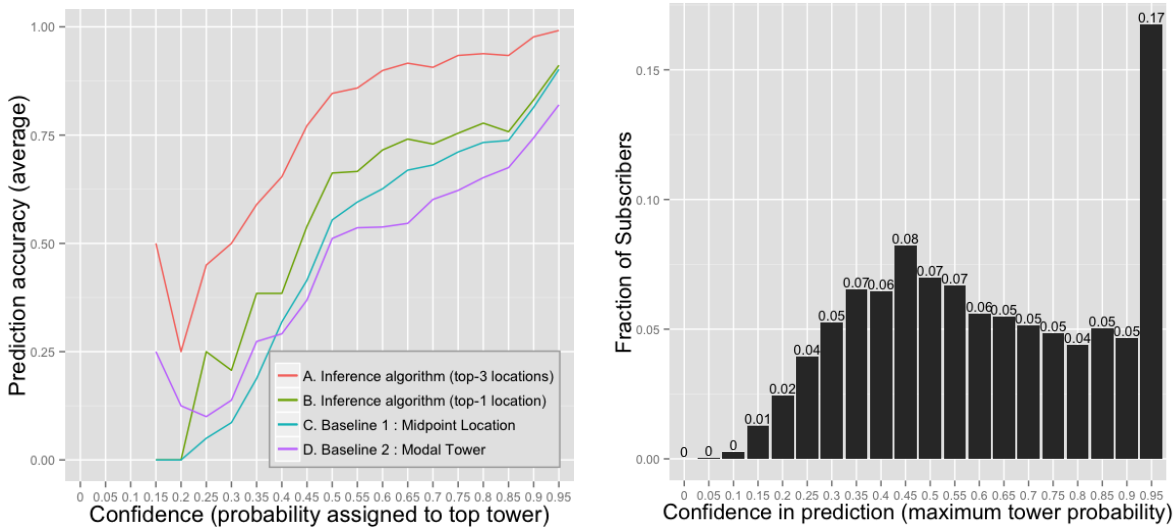


Figure 3: **Accuracy and confidence.** Figures illustrate the extent to which the accuracy is higher when prediction confidence (probability assigned to top tower) is higher. **Left:** Direct relationship between accuracy and confidence for Top Match, Top-3, and baseline models. **Right:** Distribution of predictions by confidence. A large number of predictions have high confidence, but a significant mass exists below 75% confidence.

An advantage of our probabilistic algorithm is the fact that each predicted location has an associated confidence. As can be seen when comparing the results for the random sample and the sample of active subscribers in Table 2, predictions are considerably more accurate when the location is predicted with high confidence. This relationship is further explored in Figure 3a. Here, each subscriber is assigned to a confidence bin based on the maximum predicted probability from the inference algorithm. The average performance of each bin is then plotted on the y-axis, for each of the four methods. All methods steadily improve as the confidence increases. The distribution of these confidences is given in Figure 3b; while the majority of predictions are made with high confidence, a significant number of predictions are made with less than 50% confidence.

## 5.3 Performance improvements and heterogeneity

As we have seen in Figure 3, the model performs significantly better on queries where the confidence in the prediction is high. We now investigate in greater detail the types of queries and types of individuals for whom predictions are more accurate.

### *5.3.1 Active subscribers*

An important feature of our algorithm is the fact that it is still relatively accurate on populations who are not active users of the mobile phone network. At the same time, and perhaps unsurprisingly, we find that predictions for more active subscribers are considerably more accurate. Figure 4a provides direct evidence of this correlation (significant at $p < 0.05$). To construct the upper portion of Figure 4a, we divide the population of subscribers into bins based on the number of calls made in the previous month, randomly sample 1,000 subscribers from each bin, then report the predictive accuracy for each bin. The histogram on the bottom half of the figure shows the call distribution of the at-large population.

Roughly 90% of the population makes 20 or fewer calls per day, and it is this large subset for whom predictions are the least accurate. Performance steadily increases as the sparsity in the data decreases, to the point where our predictive accuracy is again close to 100% for the limited subset of the population making 100 or more calls per day. Unsurprisingly, this population of high-activity subscribers is also the population for whom the algorithm reports the highest confidence.

### *5.3.2 "Difficult" predictions*

While the locations of some *individuals* are sometimes inherently challenging to predict, we also find that the performance of our algorithm depends heavily on the type of *query* being tested. In particular, when the start and end points are separated by either a large time span or a large spatial distance, the accuracy of the pre-

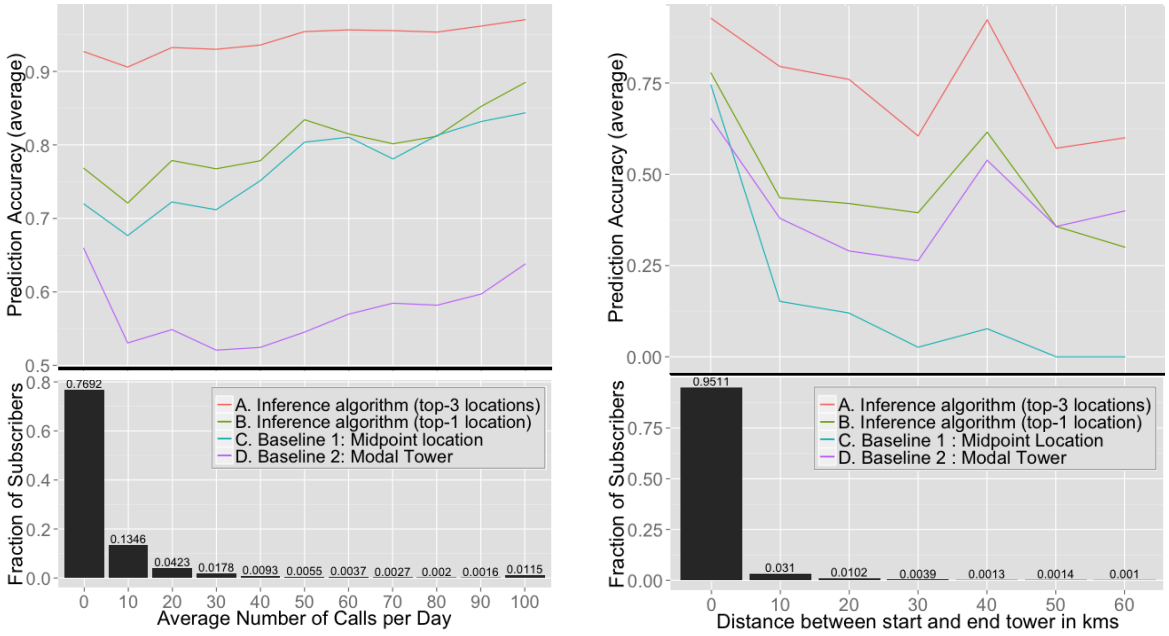


Figure 4: **Performance improvements and heterogeneity.** Predictive accuracy improves for different types of individuals and queries. The top half of each figure shows the accuracy for the Top Match and Top-3 models as well as the two baselines; the bottom half of the figure shows the fraction of the population in each respective bin. **Left:** For individuals who make a large number of calls, accuracy is higher. **Right:** Performance is better when the geographic distance between the start and end towers is smaller.

diction degrades. In Figure 4b, we show the relationship between predictive accuracy and the geographic distance between the start and end points. While accuracy is highest for the vast majority of queries where the geographic distance is less than 10 kilometers, performance drops quickly for the small number of queries with a larger distance. We find qualitatively similar results when we plot the predictive accuracy against the length in hours between the start and end sightings (results not shown).

These results are quite intuitive. When an individual travels a long distance, there are often several routes that could be taken, and even along a fixed path the speed of travel might vary. In addition, because long trips are relatively rare, there are fewer individuals in "the collective" on which to base predictions. Likewise, when a longer interval of time passes between subsequent observations – as is common with mobile phone use in many developing economies – there is inherently more uncertainty in the individual's location between sightings.

## 6. DISCUSSION AND CONCLUSIONS

We have presented a model and algorithm for predicting the unknown locations of individuals based on collective, historical patterns of travel. Testing this algorithm on a large dataset of mobile phone call detail records, we find that our "Top Match" algorithm significantly out-performs a baseline based on the subscriber's most frequently visited location, and achieves modest improvements over a method that linearly interpolates the midpoint between other known locations. Generalizing to a "Top-$N$" prediction paradigm, our algorithm is over 90% accurate with $N = 3$ (we avoid direct comparison of the Top-$N$ method to the baselines as the baselines would similarly perform better if they were allowed multiple predictions).

In the sections above, we have highlighted some of the primary advantages of this particular method: it performs well on populations typical of developing countries, where the median subscriber makes far fewer transactions than the median subscriber in the global north; it is non-parametric, and does not rely on secondary knowledge of roads or other geographical features (though as we see in Figure 2, it is capable of recovering these features from the call data); and the predictions of the algorithm have associated probabilities that indicate the relative confidence of the prediction. Important to the effectiveness of this approach is the fact that the algorithm incorporates information on trajectories of the collective, for as noted above, data on any single individual is likely to be sparse. Indeed, in our dataset, the median query has zero matches from the individual's past trajectories but 3,274 from the collective (the respective means are 1.7 and 4649). Thus, while each individual match is very heavily weighted ($\alpha = 0.99$), for any single query the collective influence is still very strong. Perhaps more importantly, this method allows for predictions even in the large number of instances when no matching information at the individual level can be found.

Along with these strengths, this method has several limitations that deserve mention, and which may provide fertile ground for future work. First, it should be stated explicitly that this model works well precisely because it relies on collective behavior when data on any given individual is scarce. Thus, it uses a "wisdom of the crowd" approach that may not be appropriate for certain individuals or trajectories. For instance, if a given individual likes to take routes or shortcuts that are unknown to the collective, unless she has used her phone consistently on those routes in the past the algorithm is likely to incorrectly predict that she is taking the road more travelled by her peers.

Second, this model is designed to predict "normal" behavior, and is therefore not well suited to understand or model behavior in times when the entire collective deviates from historical patterns of movement. For instance, on days when unexpected events occur, such as natural disasters or irregular holidays, this method will ex-

hibit a strong bias toward predictions more consistent with regular patterns of travel.

Third, unlike some of the related work discussed in the introduction, where the goal is to predict a future unknown location given only past trajectories, the method we discuss is designed to predict missing locations when both the start-point and the end-point are known – a problem that is undoubtedly more tractable. The prediction of future locations has important applications in developing countries, and slight adaptations of our framework may work well in such contexts.[2] However, our focus is on the reconstruction of historical trajectories because many of the motivating applications discussed in the introduction have this precise need, from measuring migration and mobility to modeling the spread of infectious diseases.

Lastly, issues of privacy and anonymity, while not the topic of this paper, cannot be ignored altogether. The premise of this project – that there can be humanitarian value in being able to predict the unknown location of an individual when she is "off the grid" – can easily be flipped into a scenario in which a nefarious actor might use such techniques for ill. We take some solace in the idea that in our application the accuracy can never exceed the granularity of a mobile phone tower, but also realize that in principle the model could be applied to data of higher resolution. In the end, however, we do believe that the possible gains from judicious use of these methods outweigh the possible costs of misappropriation.

In future work, we see many immediate areas for improvement and extension. In particular, the rudimentary parameterization of the similarity metrics as binary functions (Section 3.2) is a crude approximation that should be made more flexible. Allowing for a continuous measure of similarity between individuals instead of a parameter that is weighted $\alpha$ for the same individual and $(1-\alpha)$ for different individuals, would be a natural next step. Such similarity could be assessed based on direct connections between individuals or by similarity of past travel patterns. Likewise, the algorithm would likely benefit from a more flexible measurement of the similarity in the temporal features of two trajectories. These shortcomings and obvious next steps notwithstanding, we hope that this method can serve as the basis for algorithms better suited to modeling the mobility of heterogeneous populations in resource-constrained environments.

# 7. REFERENCES


[1] K. W. Axhausen and T. Gärling. Activity-based approaches to travel analysis: conceptual frameworks, models, and research problems. *Transport Reviews*, 12(4):323–341, 1992.

[2] L. Bengtsson, X. Lu, A. Thorson, R. Garfield, and J. Von Schreeb. Improved response to disasters and outbreaks by tracking population movements with mobile phone network data: a post-earthquake geospatial study in haiti. *PLoS medicine*, 8(8):e1001083, 2011.

[3] J. Blumenstock and N. Eagle. Mobile divides: Gender, socioeconomic status, and mobile phone use in Rwanda. In *Proceedings of the 4th ACM/IEEE International Conference on Information and Communication Technologies and Development*, ICTD '10, pages 6:1–6:10, New York, NY, USA, 2010. ACM.

[4] J. Blumenstock, Y. Shen, and N. Eagle. A method for estimating the relationship between phone use and wealth.


[2]Instead of assuming knowledge of the start-point and end-point and inferring the midpoint, the adapted model would assume knowledge of $N$ sequential start-points prior to a single unknown end-point. The rest of the exposition in Section 3.2 would remain unchanged.

**Algorithm 1:** Location Interpolation - Linear

**Data**: Training Data - Location records of $N$ users
$LocationMap \leftarrow$ Table of events indexed by location and time
$UserMap \leftarrow$ Table of events indexed by userID
$GT\,(GeoTemporal\ point) \leftarrow (user, time, location)$
$query \leftarrow (u^*, t^*)$

**Result**: Probability that each location $l$ is the unknown location

Step 1: Read location records of $N$ users to memory

**foreach** *record in training records* **do**
  $GT \leftarrow$ *read* record
  insert $GT$ in $LocationMap$
  insert $GT$ in $UserMap$
**end**

Step 2: Infer unknown location of query users

`// 1. find start and finish GT for query`
get $UserRecords \leftarrow e(u^*)$ from $UserMap$

**foreach** *GT in UserRecords* **do**
  find $startGT \leftarrow (u^*, t_s^*, l_s^*)$ such that $t_s^*$ precedes $t^*$
  find $finishGT \leftarrow (u^*, t_f^*, l_f^*)$ such that $t_f^*$ follows $t^*$
**end**

`// 2. find matching users to given query`
**foreach** $t \in time$ **do**
  **foreach** $l \in locations$ **do**
    **if** $k_s(t, t_s^*) = 1$ *and* $g_s(l, l_s^*) = 1$ **then**
      $u_s \leftarrow$ set of user from $LocationMap$ indexed by $(l, t)$
      insert $u_s$ in $StartUsers$
    **end**
    **if** $k_f(t, t_f^*) = 1$ *and* $g_f(l, l_f^*) = 1$ **then**
      $u_f \leftarrow$ set of user from $LocationMap$ indexed by $(l, t)$
      insert $u_f$ in $FinishUsers$
    **end**
  **end**
**end**

$MatchedUsers \leftarrow StartUsers \cap FinishUsers$

`// 3. Calculate probability of each location`
**foreach** $u \in MatchedUsers$ **do**
  **foreach** $t \in time$ **do**
    **if** $t > t_s^*$ *and* $t < t_f^*$ **then**
      $l \leftarrow Loc(u,t)$
      $W_l \leftarrow W_l + h(u, u^*) \times k(t, t^*)$
      $totalW \leftarrow totalW + W_l$
    **end**
  **end**
**end**

**foreach** $l \in locations$ **do**
  $P(Loc(u^*, t^*) = l) \leftarrow W_l / totalW$
**end**

**Algorithm 2:** Location Interpolation - MapReduce

**Data**: Location records of $N$ users
Set of $M$ queries, where $query \leftarrow (u^*, t^*)$
$GT\ (GeoTemporal\ point) \leftarrow (user, time, location)$
**Result**: For every query in $M$, probability of each location $l$ for being the unknown location

Step 1: Find start and finish GT for each query in M

**foreach** *query in M* **do**
  get $UserRecords \leftarrow e(u^*)$
  **foreach** *GT in UserRecords* **do**
    find $startGT \leftarrow (u^*, t_s^*, l_s^*)$ such that $t_s^*$ precedes $t^*$
    find $finishGT \leftarrow (u^*, t_f^*, l_f^*)$ such that $t_f^*$ follows $t^*$
    add ($startGT$, $finishGT$) to query in $M$
  **end**
**end**

Step 2: Map(key, value) - Emit records' of matched users

$UserRecords \leftarrow$ group Training data by *user*
**foreach** $user \in N$ **do**
  sort $UserRecords$ by time
  **foreach** $GT_i$ *in* $UserRecords$ **do**
    **foreach** $GT_j$ *in* $UserRecords, \forall\, i < j$ **do**
      **foreach** *query(startGT, finishGT)* $\in M$ **do**
        **if** $k_s(t_i, t_s^*) = 1$ *and* $g_s(l_i, l_s^*) = 1$
        *and* $k_f(t_j, t_f^*) = 1$ *and* $g_f(l_j, l_f^*) = 1$ **then**
          $key \leftarrow (u^*, t^*)$
          $value \leftarrow$ set of $GT_k, \forall\, i < k < j$
          emit(*key*, *value*)
        **end**
      **end**
    **end**
  **end**
**end**

Step 3: Reduce(key, values[ ]) - Prediction of unknown locations

**foreach** *element v in values* **do**
  **foreach** *GT in v* **do**
    $l \leftarrow Loc(u,t)$
    $W_l \leftarrow W_l + h(u, u^*) \times k(t, t^*)$
    $totalW \leftarrow totalW + W_l$
  **end**
**end**
**foreach** $l \in locations$ **do**
  $P(Loc(u^*, t^*) = l) \leftarrow W_l / totalW$
**end**


*QualMeetsQuant Workshop at the 4th International IEEE/ACM Conference on Information and Communication Technologies and Development*, 2010.

[5] J. E. Blumenstock. Inferring patterns of internal migration from mobile phone call records: Evidence from Rwanda. *Information Technology for Development*, 18(2):107–125, 2012.

[6] J. E. Blumenstock and N. Eagle. Divided we call: Disparities in access and use of mobile phones in rwanda. *Information Technology and International Development*, 8(2):1–16, 2012.

[7] F. Calabrese, G. Di Lorenzo, L. Liu, and C. Ratti. Estimating origin-destination flows using mobile phone location data. *IEEE Pervasive Computing*, 10(4):36–44, 2011.

[8] E. Frias-Martinez, G. Williamson, and V. Frias-Martinez. An agent-based model of epidemic spread using human mobility and social network information. In *Privacy, security, risk and trust (passat), 2011 IEEE third international conference on social computing (socialcom)*, pages 57–64. IEEE, 2011.

[9] J. B. Gomes, C. Phua, and S. Krishnaswamy. Where will you go? mobile data mining for next place prediction. In L. Bellatreche and M. K. Mohania, editors, *Data Warehousing and Knowledge Discovery*, LNCS 8057, pages 146–158. Springer Berlin Heidelberg, Jan. 2013.

[10] M. C. González, C. A. Hidalgo, and A.-L. Barabási. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, June 2008.

[11] Z. Huang and U. Kumar. Combining call records and road data for strategic disaster response planning. 2013.

[12] Y.-J. Kim and S.-B. Cho. A HMM-based location prediction framework with location recognizer combining k-nearest neighbor and multiple decision trees. In J.-S. Pan, M. M. Polycarpou, M. Woźniak, A. C. P. L. F. d. Carvalho, H. Quintián, and E. Corchado, editors, *Hybrid Artificial Intelligent Systems*, LNCS 8073, pages 618–628. Springer Berlin Heidelberg, Jan. 2013.

[13] G. D. Lorenzo, J. Reades, F. Calabrese, and C. Ratti. Predicting personal mobility with individual and group travel histories. *Environment and Planning B: Planning and Design*, 39(5):838–857, 2012.

[14] X. Lu, E. Wetter, N. Bharti, A. J. Tatem, and L. Bengtsson. Approaching the limit of predictability in human mobility. *Scientific Reports*, 3, Oct. 2013.

[15] A. Monreale, F. Pinelli, R. Trasarti, and F. Giannotti. WhereNext: A location predictor on trajectory pattern mining. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 637–646, New York, NY, USA, 2009. ACM.

[16] C. Ratti, R. M. Pulselli, S. Williams, and D. Frenchman. Mobile landscapes: using location data from cell phones for urban analysis. *Environment and Planning B: Planning and Design*, 33(5):727–748, 2006.

[17] J. C. Scott. *Seeing Like a State: How Certain Schemes to Improve the Human Condition Have Failed*. Yale University Press, Feb. 1999.

[18] K. Shilton. Four billion little brothers?: Privacy, mobile phones, and ubiquitous data collection. *Commun. ACM*, 52(11):48–53, Nov. 2009.

[19] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, Feb. 2010.

[20] A. Wesolowski, N. Eagle, A. J. Tatem, D. L. Smith, A. M. Noor, R. W. Snow, and C. O. Buckee. Quantifying the impact of human mobility on malaria. *Science*, 338(6104):267–270, Oct. 2012.

[21] H. Xiong, D. Zhang, D. Zhang, and V. Gauthier. Predicting mobile phone user locations by exploiting collective behavioral patterns. In *2012 9th International Conference on Ubiquitous Intelligence Computing (UIC/ATC)*, pages 164–171, Sept. 2012.