City-Scale Localization with Telco Big Data

Fangzhou Zhu[1,2], Chen Luo[1,2], Mingxuan Yuan[3,*], Yijian Zhu[4], Zhengqing Zhang[4], Tao Gu[5], Ke Deng[5], Weixiong Rao[6,*] and Jia Zeng[1,2,3,*]
[1]School of Computer Science and Technology, Soochow University, Suzhou 215006, China
[2]Collaborative Innovation Center of Novel Software Technology and Industrialization
[3]Huawei Noah's Ark Lab, Hong Kong
[4]China United Network Communications Corporation Limited Shanghai Branch
[5]Computer Science, RMIT University
[6]School of Software Engineering Tongji University, China
*Corresponding Authors: Yuan.Mingxuan@huawei.com, wxrao@tongji.edu.cn, zeng.jia@acm.org

ABSTRACT

It is still challenging in telecommunication (telco) industry to accurately locate mobile devices (MDs) at city-scale using the measurement report (MR) data, which measure parameters of radio signal strengths when MDs connect with base stations (BSs) in telco networks for making/receiving calls or mobile broadband (MBB) services. In this paper, we find that the widely-used location based services (LBSs) have accumulated lots of over-the-top (OTT) global positioning system (GPS) data in telco networks, which can be automatically used as training labels for learning accurate MR-based positioning systems. Benefiting from these telco big data, we deploy a context-aware coarse-to-fine regression (CCR) model in Spark/Hadoop-based telco big data platform for city-scale localization of MDs with two novel contributions. First, we design map-matching and interpolation algorithms to encode contextual information of road networks. Second, we build a two-layer regression model to capture coarse-to-fine contextual features in a short time window for improved localization performance. In our experiments, we collect 10^8 GPS-associated MR records in the centroid of Shanghai city with 12×11 square kilometers for 30 days, and measure four important properties of real-world MR data related to localization errors: stability, sensitivity, uncertainty and missing values. The proposed CCR works well under different properties of MR data and achieves a mean error of $110m$ and a median error of $80m$, outperforming the state-of-art range-based and fingerprinting localization methods.

1. INTRODUCTION

In the past decade, location-based service (LBS) has gained skyrocketing usage with big business value because location gives context to current spatiotemporal events such as working, shopping, payment, navigation, social networking, transportation and security. Commercial examples of location context-aware applications include Baidu Map, Alipay and Uber in China. In the age of context, the global positioning system (GPS) plays an important role in locating accurately an mobile device (MD) outdoor with around $\pm 10m$ errors based on satellite networks. Unfortunately, GPS has a few weaknesses: 1) energy-consuming, 2) unavailable in many MDs, 3) requiring line-of-sight to the satellites (e.g., GPS degrades quickly indoors or underground), and 4) being often turned off for some private reasons. Therefore, localization methods using measurement report (MR) data from telco networks (GSM, CDMA and LTE) have attracted intensive research interests in telecommunication (telco) industry [4, 7, 24, 22, 14, 15, 16, 2, 5].

When compared with GPS, the MR-based positioning systems have the following advantages [22, 14, 1]: 1) energy-efficient, 2) available in most mobile phones or devices, 3) better network coverage and workable indoors and underground, and 4) active when making calls or mobile broadband (MBB) services. Hence, city-scale localization with telco big data is a good complement to GPS for a better LBS experience without extra overhead. However, it is still challenging to achieve a comparable localization performance of GPS because the current localization error using telco networks has a very large range $50 \sim 1000m$ in different situations [24, 14, 16]. Similarly, city-scale localization has also been investigated using the WiFi network [8], which is another type of wireless sensor network (WSN) for positioning tasks with a long history. Indeed, integration of MD sensor, WiFi and telco network data can achieve a high outdoor localization accuracy [1]. For simplicity, this paper focuses on deploying a more accurate MR-based positioning system in the telco big data platform [23, 13, 12, 17, 19]. Such a direction is promising because the next-generation (e.g., 5G) telco network will have denser structures (smaller cell sizes) which may improve the localization performance.

The real-world MR data have four important properties affecting the localization accuracy: stability, sensitivity, uncertainty and missing values. First, due to multipath propagation, non-line-of-sight propagation and multiple access interference [4], MR data often change temporally for the same location and MD. If the temporal stability of MR data is low, the positioning system should adapt its parameters to the large variations of MR data for high localization accuracy. Second, the spatial sensitivity of MR data measures whether a small change of moving location will cause a salient change of MR data. For example, if an MD moves $10m$ and its corresponding MR data do not change, the positioning system cannot differentiate the location points within $10m$ range. Third, most MDs will not connect to the closest sector or base station (BS) because the signal arriving at the BS from the MD is reflected or diffracted and takes a longer path than the direct path. The high connection uncertainty will deteriorate the localization accuracy.

CIKM '16, October 24–28, 2016, Indianpolis, IN, USA.

http://dx.doi.org/10.1145/2983323.298334

Table 1: Abbreviations & Meanings.

AGPS	assisted global positioning system
AOA	angle of arrival
BS	base station
BSS	business supporting system
CCR	context-aware coarse-to-fine regression model
CDF	cumulative distribution function
CDMA	code division multiple access
CI	cell ID
DPI	deep package investigation
GPS	global positioning system
GSM	global system for mobile communications
IMSI	International Mobile Subscriber Identity
LAC	location area code
LBS	location-based service
LTE	long-term evolution
MBB	mobile broadband
MD	mobile device
MR	measurement report
OSS	operation supporting system
OTT	over-the-top
RF	random forest
RNC	radio network controller
RSSI	radio signal strength indicator
SGSN	serving GPRS support node
TDOA	time difference of arrival
TOA	time of arrival

Usually, localization accuracy of MDs connecting to its closest BS is much higher than that of MDs without doing so. Finally, not all MR data have radio signal strength indicator (RSSI) values for 6 sectors [24]. More than 50% of real-world MR records contain RSSI with only 1 or 2 BSs or sectors. The missing values of MR data also deteriorate the localization accuracy because triangulation or trilateration techniques cannot be performed [24]. To the best of our knowledge, previous studies rarely investigate the four important properties of big MR data.

For the better MR-based positioning system, this paper makes two main contributions. The primary contribution is an empirical measurement over four important properties of big MR data, which provides practical guidance on how to update the localization models and determine the lower limit of localization errors. As the second contribution, we deploy a machine learning-based positioning system from telco big data called context-aware coarse-to-fine regression model (CCR) for MD localization in telco networks. Data-driven predictive modeling generally includes constructing useful feature vectors (aka predictor variables) and training good regression and classification models (aka predictors) with training data of labeled features. After training, we aim to predict the location coordinates without location labels, where training and test data have no overlap in time intervals. The regression model can be updated periodically by new coming data for a high accuracy.

First, we find that many LBS over-the-top (OTT) applications from different MDs have automatically accumulated a large number of GPS data in telco networks, which enables automatically labeling the MR data with the corresponding ground truth GPS coordinates (as GPS is the most accurate sensor for outdoor localization till now). This strategy avoids the high cost of additional war driving to collect training data, where cars drive [22, 14] or individuals walk [16] the area of interest continuously scanning for BSs or sectors and recording the local area code (LAC) and Cell ID, RSSI, and GPS location. Second, we use the map-matching algorithm [18] to match low-sampling-rate GPS data from various MDs to buildings and road networks. This can further improve the accuracy of ground truth GPS coordinates by contextual information. To link with the corresponding high-sampling-rate MR data by time stamps, we use the most-frequent-path algorithm [20] to interpolate mapped GPS data with different sampling rates. Thus, we obtain automatically lots of GPS-associated MR data encoding structural information of road networks as training data. Third, we train a two-layer random forest (RF) [3] regression model that builds the functional mapping from MR data to the ground truth GPS coordinates. In the first layer, we input the labeled MR-based feature vectors to train the regression model and output the predicted GPS coordinates of the training data as the coarse location features. In the second layer, we design the fine-grained contextual features including velocity, direction and change of sectors in a time window as input based on the coarse location features produced by the first layer regression model. More specifically, the output of the first layer regression model as well as a group of contextual fine-grained features are used as the input of the second layer regression model to calculate the accurate position. With the cascaded architecture, CCR can capture coarse-to-fine contextual information for a higher localization accuracy. The evaluations on real-world big MR data confirms that CCR outperforms significantly the current state-of-art localization methods such as the range-based Bayesian inference [24], the grid-based fingerprinting [14], and the map-aware sequential matching [22, 16]. To summarize, we make the following contributions in this work:

- We automatically obtain big GPS-associated MR data by integrating telco big data, which solves the high-cost labeling problem in previous localization methods on telco networks. Through measuring stability, sensitivity, uncertainty and missing values of MR data, we show some practical guidance on building reliable MR-based positioning systems.
- We deploy the CCR model for MR-based localization. First, CCR uses map-matching and interpolation algorithms to encode structural information of road networks. Second, CCR has a cascaded two-layer structure, where the coarse location output by the first layer is processed to be the fine-grained input feature to the second layer to capture contextual information in a short time window. We conduct extensive experiments on 10^8 GPS-associated MR records from 1.8×10^4 MDs to compare CCR with the state-of-art solutions under different MR data properties.

Table 1 summarizes some important abbreviations appearing in this paper. Section 2 describes the CCR localization model, which includes map-matching, most-frequent-path interpolation and a two-layer random forest regression model. Section 3 measures the real-world MR data on their four properties: stability, sensitivity, uncertainty and missing values. Section 4 compares the localization performance between CCR and the state-of-the-art methods. Also, it discusses some practical issues in deploying CCR according to different properties of MR data. Section 5 reviews the related techniques on localization in telco networks. Finally, Section 6 draws conclusions and envisions future work.

2. CCR LOCALIZATION MODELS

Figure 1 shows the hierarchy of a typical UMTS (Universal Mobile Telecommunications System) telco network and describes how MR data are generated. The network in an urban area is divided into several large regions denoted by location area code (LAC). Each LAC contains several radio network controllers (RNC), which is composed of several location-associated base stations (BS) or cell towers. Each BS is associated with several sector antennas (usually 3 for three directions) denoted by cell ID (CI). Each cell uses a different set of frequencies from neighboring cells, to avoid interference and provide guaranteed bandwidth within each cell. Before an

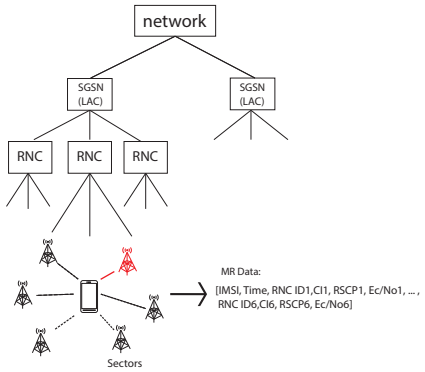

Figure 1: The overview of MR data generation procedure.

Table 2: A detailed example of the MR record.

Field	Example	Field	Example
MRTime	2015/12/16 08:00:00.000	IMSI	************058
SRNCID	2350	BestCellID	31171
SRNTI	4753	RAB	0,1,2,3,4,5
Event	0,1,2,3,5,8,-999	Delay	3
UE_TXPower	20	RNCID_1	2350
CellID_1	31171	EcNo_1	-6.5
RSCP_1	-85	RTT_1	1037
UE_Rx_Tx_1	1033	RNCID_2	2350
CellID_2	31171	EcNo_2	-6.5
RSCP_2	-85	RTT_2	1037
UE_Rx_Tx_2	1033	...	...
...	...	...	...
RNCID_6	2350	CellID_6	31171
EcNo_6	-6.5	RSCP_6	-85
RTT_6	1037	UE_Rx_Tx_6	1033

MD connects to the network, it measures parameters of radio signal strength with nearby candidate sectors and connects to the best sector with the strongest signal strength. Generally, MR data contain measurement parameters such as the radio signal strength indicator (RSSI) for signal strength of 6 nearby sectors. Figure 1 shows an example of MR record, including user ID IMSI (International Mobile System Identity), time stamp, six BS ID (RNC ID or LAC and CI for CDMA network; enodeB ID for LTE network), and six RSSI = RSCP − EcNo (RSCP is Received Signal Code Power and EcNo is Energy per Bit to Noise Power Density). Table 2 shows a detailed example of the MR record. RNCID_x and CellID_x represent the RNC ID and Cell ID to identify a unique sector in the location-associated BS. RTT_x is the round-trip-time [24] of the signal from the MD to the sector. UE_Rx_Tx_x is the time that the MD receives and transmits UE (user equipment) signal. The MR record usually contains values from 6 nearby sectors. But more than 50% MR records have missing values with no more than two sectors [24] in the real-world telco network. Besides RSSI, MR data can also have other radio parameter measurements such as TOA/TDOA and AOA [4, 14]. Without loss of generality, we use only RSSI as well as engineering parameters of location-associated BSs to build the MR-based positioning system.

Figure 2 shows the data flow of the context-aware coarse-to-fine regression (CCR) localization model in telco big data platform, where (a) shows the OTT GPS locations (red dots) generated by LBSs (GPS data often have low sampling rate) and (b) shows two examples of MR data generated by call or MBB services (high sampling rate). The red dots are the real GPS locations of the corresponding MR data (the locations to predict using MR data). In the training phase, CCR learns model parameters based on a large number of training samples such as [RSSI feature vector, GPS coordinate].

Algorithm 1: Training Data Acquisition.

Input: d_{mr} - MR data, d_{ott} - OTT GPS data, map - map, t_g - time granularity to do interpolation
Output: $TrainingData$

1 $urls = DPI(d_{ott})$;
2 raw_urls_{gps} = Extract_GPS($urls$);
3 $trajs_{url}$ = Cut_Trajectories(raw_urls_{gps});
4 $urls_{gps} = \{\}$;
5 **for** *each trajectory* t *in* $trajs_{url}$ **do**
6 $road_t$ = Map_Matching(t, map);
7 **for** *each adjacent points* p_{i-1} *and* p_i *in* $road_t$ **do**
8 $path_{i-1,i}$ = Most_Frequent_Path(p_{i-1}, p_i);
9 $urls_{gps} = urls_{gps} \cup$ Interpolation($p_{i-1}, p_i, path_{i-1,i}, t_g$);
10 $TrainingData$ = match($urls_{gps}, d_{mr}$);
11 Return $TrainingData$;

In the prediction phase, CCR maps RSSI feature vectors of an MD to the predicted positions and trajectories as shown in Figure 2 (c). Figure 2 (d) shows the predicted location (blue dots) when compared with the ground truth GPS location (red dots), where the localization error is the distance between the predicted location and the ground truth GPS location. Although the MR records used in this paper are generated by UMTS, the proposed CCR model can be also used for MR records generated by both GSM or LTE. There are two main steps in running CCR localization model:

- Training data acquisition: we automatically associate MR records with corresponding GPS coordinates (longitude, latitude). We use map-matching [18] and most-frequent-path [20] interpolation algorithms to fulfil this step in Subsection 2.1.
- CCR training and prediction: we train a two-layer random forest [3] regression model with coarse-to-fine labeled feature vectors, and predict location by mapping feature vectors based on trained regression model. Details of this step are described in Subsection 2.2.

2.1 Training Data Acquisition

Algorithm 1 shows the overall procedure to obtain the training data. Since the MR and OTT GPS data from OSS are automatically stored in telco big data platform, the labeling of MR data can be implemented without bringing any extra burden to telco networks. First, we use DPI (deep package investigation) tools to extract the $urls$ in OTT data (line 1), where many $urls$ have GPS coordinates. We select urls with high qualities (eg. taxi services) to ensure stable and continuous trajectories along road networks. Second, we extract the GPS coordinates from these $urls$ (line 2) and cut the long GPS trajectory into segments when the interval of time stamps is larger than 5 minutes or two GPS distance is larger than 1 kilometer (line 3). Because the sampling rates of MR records and GPS data are quite different, it will cause lots of unmatched MR records. For example, the sampling rate is 8 seconds per record in MR and 60 seconds per record in GPS data. To address this problem, we propose a novel solution with map-matching [18], most-frequent-path [20] interpolation to increase the sampling rate of GPS data. Third, we match the GPS coordinates in a short trajectory with the road network in the map (line 6). For each pair of adjacent matched points (the projected position of a GPS coordinate to its corresponding road), we compute the most-frequent-path between these two points and do uniform interpolation along this frequent path (lines 8 and 9). Finally, we match the interpolated GPS data with the MR data by IMSI and time stamp in a sliding window. The functions between line 5 and line 9 in Algorithm 1 are described in a single thread form to demonstrate the design logic. In our distributed Hadoop/Spark system, they are implemented in parallel.

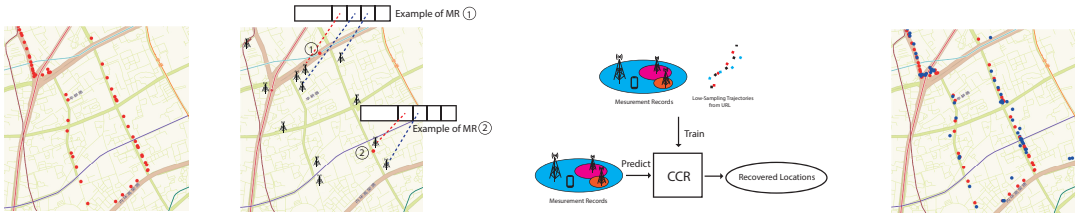

Figure 2: The data flow of CCR model. (a) shows the OTT GPS locations (red dots) generated by LBSs (low sampling rate). (b) shows two examples of MR data generated by call or MBB services (high sampling rate). The red dots are the real GPS locations of the corresponding MR data (the locations to predict using MR data). (c) CCR takes as input of OTT GPS (training labels) and MR (training features) data to estimate model parameters for training. In prediction, the trained CCR can predict the location given only MR data as input. (d) shows the predicted location (blue dots) when compared with the ground truth GPS location (red dots), where the localization error is the distance between the predicted location and the ground truth GPS location.

Algorithm 2: Map-matching.

Input: t - a trajectory from OTT, map - map
Output: $road_t$ - the revised trajectory by mapping each point on road

for *each point* p_i *in* t **do**
  calculate the candidate road set R_i for p_i with probabilities estimated from distances;
for *each adjacent points* p_{i-1} *and* p_i **do**
  calculate the transmission probabilities between the roads in R_{i-1} and R_i;
calculate the road sequence with the largest probability using HMM (Hidden Markov Model);
return the $road_t$ by setting the GPS of each point as its projected position on corresponding road;

Algorithm 3: Most-frequent-path Interpolation.

Input: p_{i-1}, p_i - two adjacent points in map-matched trajectory $road_t$, $path_{i-1,i}$ - the most frequent path between p_{i-1} and p_i, t_g - time granularity to do interpolation
Output: $(p_{i-1}, c_1, c_2, ..., p_i)$

$n = \lceil \frac{p_i.time - p_{i-1}.time}{t_g} \rceil$;
$length = \sum_{roadr \in path_{i-1,i}} length(r)$;
$v = \frac{length}{p_i.time - p_{i-1}.time}$;
$result = (p_{i-1})$;
for *j = 1 to n* **do**
  $c_j.position = p_{i-1}.position + v \times j \times t_g$ along $path_{i-1,i}$;
  $c_j.time = p_{i-1}.time + j \times t_g$;
  $result = result \cup c_i$;
$result = result \cup p_i$;
return $result$;

Algorithm 4: CCR Training Procedure.

Input: $TrainingData$ - the GPS-associated MR data
Output: m_{loc} - localization models for longitude and latitude

for *each record* r *in* $TrainingData$ **do**
  $r.feature$ = [coarse features in Table 3];
$m_{loc,layer1}$ = a RF model using coarse features;
for *each record* r *in* $training_{data}$ **do**
  $r.feature$ = [output of $m_{loc,layer1}$, coarse features in Table 3, fine-grained features in Table 4];
$m_{loc,layer2}$ = a RF model by adding fine-grained features;
return m_{loc};

Table 3: Coarse Features.

Features	Description
rssi	Received Signal Strength Indication
rscp	Received Signal Code Power
ecno	Ratio of energy per modulating bit to the noise spectral density
rncid	RNC id
ci	cell id
lon	longitude of a sector
lat	latitude of a sector
id	unique id for a BS
height	height of a BS
azimuth	azimuth of a BS
mdtilt	Mechanical Down Tilt
edtilt	Electrical Down Tilt
bs_type	type code of a BS (such Metrocell, Microcell, etc.)
company	producer of a BS (such as HUAWEI, Nokia)
n_sector	# associated sectors
n_bs	# associated BSs

Using Algorithm 1, we obtain the fine-grained GPS-associated MR data for training CCR. Experiments show that map-matching and most-frequent-path interpolation increase the training data quality for better localization performance in section 4.

Algorithm 2 shows how to match each GPS point t in the generated trajectory set $trajs_{url}$ to map using hidden Markov models (HMMs) [18]. First, we select the candidate road set for each point t. The probability of selecting a candidate road for a point t is estimated by their closest distance. The transition probabilities between the roads in the candidate road set of two adjacent points are estimated from historical data. Second, we use the Viterbi algorithm to infer a road sequence with the largest probability from the candidate road set. This road sequence is the map-matching result. The GPS point t can be replaced by its projected position on the corresponding road. Finally, we obtain a new map-matched trajectory $road_t$ encoding road network structures.

Algorithm 3 shows the interpolation procedure to increase the sampling rate of GPS data. We uniformly insert new GPS points between two adjacent GPS points in $road_t$ with constant small time step such as 2 seconds. The points are interpolated on the road along which we assume the MD moves with the uniform velocity. We use most-frequent-path (MFP) algorithm [20] to infer the road sequence between two adjacent points in $road_t$. MFP is the path that most MDs move along, which is a type of useful collaborative knowledge in the GPS data. The new GPS points are interpolated along the calculated road sequence. After interpolation, we match the new GPS position data and MR data with the same sampling rate and obtain the training data. Using interpolation, the volume of training data can be increased around 20 times and the localization error can be reduced more than 10% confirmed in Section 4.

2.2 CCR Training and Prediction

Algorithm 4 shows the training procedure of CCR. We train a two-layer regression model with the GPS-labeled MR data. The first layer regression model takes as input 258 dimensional coarse features partly in Table 3, such as RSSI and the related engineering parameters of BSs (e.g., antenna height, azimuth, longitude, latitude and etc), and predict GPS location of each training sam-

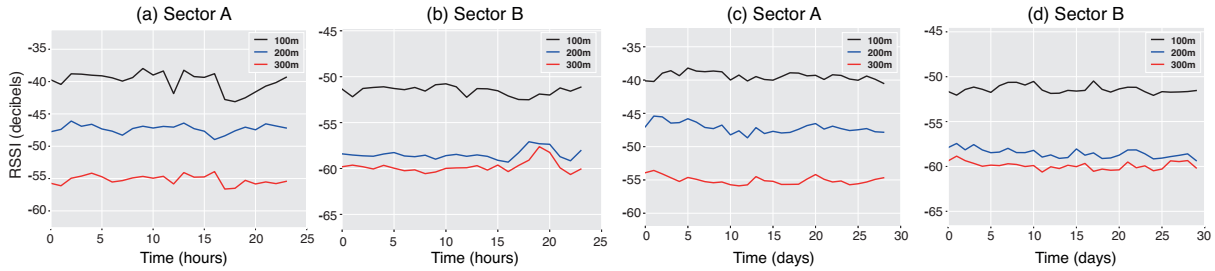

Figure 3: Stability measurement over time (hours and days) of MR data for two typical sectors A and B.

Table 4: Fine-grained Features.

Features	Description
pred_lon	p_i's longitude from the output of the first layer
pred_lat	p_i's latitude from the output of the first layer
last_distance	distance between p_{i-1} and p_i
last_direction	direction from p_{i-1} to p_i
last_speed	average speed from p_{i-1} to p_i
last_time_gap	time gap between p_{i-1} and p_i
last_lon	p_{i-1}'s longitude from the output of the first layer
last_lat	p_{i-1}'s latitude from the output of the first layer
next_distance	distance between p_i and p_{i+1}
next_direction	direction from p_i to p_{i+1}
next_speed	average speed from p_i to p_{i+1}
next_time_gap	time gap between p_i and p_{i+1}
next_lon	p_{i+1}'s longitude from the output of the first layer
next_lat	p_{i+1}'s latitude from the output of the first layer
last_heading_change_angle	delta angle between p_{i-1} and p_i
last_speed_change	delta speed between p_{i-1} and p_i
next_heading_change_angle	delta angle between p_i and p_{i+1}
next_speed_change	delta speed between p_i and p_{i+1}
angle2azimuth	the angle between p_i and main BS's azimuth

Algorithm 5: CCR prediction procedure.

Input: d'_{mr} - MR data of a device, m_{loc} - localization model, et_{bs} - engineering table of BSs
Output: t' - a recovered trajectory

$t' = []$;
for *i from 1 to* $|t'|$ **do**
  if $i == 1$ **then**
    $pos_i(lon, lat) = m_{loc}(d'_{mr}[i, i, i+1], et_{bs})$;
  if $i == |t'|$ **then**
    $pos_i(lon, lat) = m_{loc}(d'_{mr}[i-1, i, i], et_{bs})$;
  else
    $pos_i(lon, lat) = m_{loc}(d'_{mr}[i-1, i, i+1], et_{bs})$;
  t'.add(($t'[i]$.time, $pos_i(lon, lat)$));
return t';

Table 5: Statistics of GPS-associated MR Dataset.

Type	Number
Time	30 days
Area	12×11 square killometers
Data blocks	4×4 blocks
Number of BSs	$2,697$
Number of Sectors	$18,431$
Number of IMSIs	$17,699$
Number of Trajectories	$2,181,990$
Number of GPS-associated MR records before map-matching and interpolation	$4,749,150$
Number of GPS-associated MR records after map-matching and interpolation	$103,605,330$

ple. Based on the predicted positions, we add 34 dimensional fine-grained contextual feature vectors partly in Table 4, velocity, moving direction, moving distance, velocity change, moving direction change, to train the second layer regression model. These fine-grained features help to calculate longitude/latitude more accurately. We confirm that this two-layer design performs much better than the single-layer regression model in Section 4.

We choose RF [3] in each layer to do regression. The optimization objective of RF for regression is

$$S = \sum_{C \in leaves(T)} \sum_{i \in C} (y_i - m_c)^2, \quad (1)$$

where y_i is the target longitude/latitude, T is a tree in RF and $m_c = \frac{1}{n_c} \sum_{i \in C} y_i$. This optimization function S guides the algorithm to cut the feature space to small subspaces with similar longitude/latitude values. The feature engineering and RF regression model are hand coded in Spark, which is based on Hive/Spark SQL and the parallel RF algorithm [13]. Note that distributed RF is scalable to big data without communication cost among machines because each regression tree is trained independently.

Algorithm 5 shows the prediction procedure of CCR without training data. In working scenarios, CCR maps the RSSI feature vector of an MD to the predicted location coordinate. By predicting a sequence of location coordinates, the trajectory of the MD is recovered on the road networks.

3. DATASETS AND MEASUREMENTS

Table 5 shows the datasets for the measurements and experiments. From the telco big data platform, we extract GPS-associated MR records for a month, which covers 12×11 square kilometers in centroid of Shanghai, China. We divide equally the entire area into $4 \times 4 = 16$ blocks. For each block, we will train a CCR localization model. The number of location-associated BSs in the region of interest is 2697. The number of unique IMSI MDs is 17699, which covers a variety of mobile device types. This can be used to evaluate the robustness of CCR for different types of MDs. The number of trajectories is more than 2 millions. The number of MR records which can be associated with GPS position is around 5 million by directly liking without map-matching and interpolation. After map-matching and interpolation in Subsection 2, the number of GPS-associated MR records increases to 10^8, which is big enough for evaluate the deployed CCR localization model.

3.1 Stability

Figure 3 shows the change of RSSI over time (hours and days) at the same range of locations $\{100m, 200m, 300m\}$ of two typical sectors A and B. Generally, the smaller distance to the sector will lead to stronger RSSI values. In $100m$ RSSI curve (black line), the point is the average RSSI value of all GPS points at $100m$ around the sector during each hour or each day. This temporal stability measurement shows that even if each an MD stays at the same position, it still receives different RSSIs from the sector at different time stamp. We see that the fluctuation over days (c) or (d) is smaller than that over hours (a) or (b) in Figure 3. Through statistical analysis of all sectors in the dataset, the $|\Delta$RSSI$|$ over days

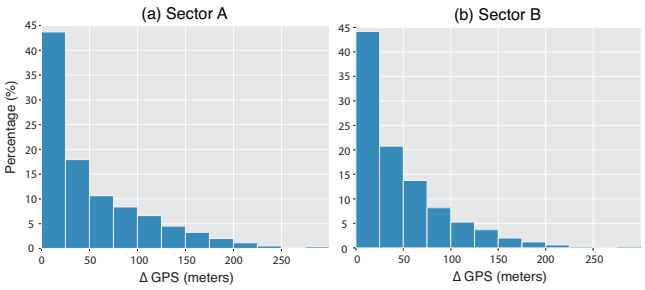

Figure 4: Sensitivity measurement over space of MR data (using ΔGPS when $|\Delta\text{RSSI}| \leq 0.7$ for two sectors A and B.

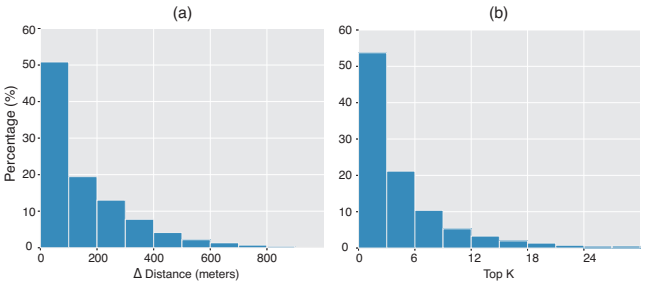

Figure 5: Uncertainty measurement of all MR data. (a) shows the distribution over ΔDistance between the connected BS and the closest BS. (b) shows the distribution over the top K closest BSs.

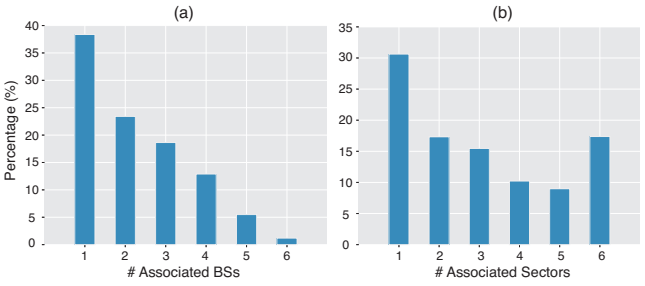

Figure 6: The distribution of MR data over the number of associated BSs (a) and sectors (b).

(e.g., in (c) and (d)) is often less than 0.7, which implies that the movement with RSSI change less than ± 0.7 is indistinguishable because of the temporal instability. Suppose that an MD is at $100m$ of the sector A with RSSI -40. If it moves to $120m$ of the sector with RSSI -40.7, the localization model cannot differentiate this $20m$ movement distance because the location $100m$ can also have RSSI -40.7 due to temporal fluctuations. The key inspiration is that if the localization model is not updated by the new training data, the model may fail to capture the temporal change of RSSI for bad localization performance.

Figure 3 (a) and (b) show that the change of RSSI over hours is very large from time stamp $17:00$ to $19:00$, which is the work-to-home peak hour. To enhance the localization performance, we need to update models for each hour to capture the temporal variance of RSSI. But in practice, we have insufficient and uneven training data for each hour to learn new models so that we update partial CCR models (build new regression trees in random forest by new coming training data) per day rather than per hour. Our experiments in Section 4 confirm that update models by new coming data will enhance the overall localization performance. Although Figure 3 shows temporal stability for two typical sectors, most other sectors have similar RSSI change patterns over time.

3.2 Sensitivity

Figure 4 shows the spatial sensitivity measurement of MR data for two typical sectors A and B. For the same hour, from all the RSSI values of the same sector, we select those pairs if $|\Delta\text{RSSI}| \leq 0.7$, where 0.7 is the temporal fluctuation threshold determined by Figure 3. Then, we plot the histogram of the earth distance ΔGPS of those pairs in Figure 4,

$$\Delta\text{GPS} = ||\text{GPS}_{\text{RSSI}_1} - \text{GPS}_{\text{RSSI}_2}||_{\text{earth}}. \quad (2)$$

As we discussed, this distance cannot be differentiated by the localization models. We see that more than 50% of pairs have the distance $\geq 25m$, which implies these points cannot be recognized within the radius $25m$. The spatial sensitivity is consistent with the systematic lower limit of the localization error. From these two typical sectors, the RSSI data cannot provide the median localization error less than $\pm 25m$ or around $50m$. Although Figure 4 shows the sensitivity measurement for two typical sectors, most other sectors have similar distance ΔGPS histograms when $|\Delta\text{RSSI}| \leq 0.7$. From another perspective, we assume that $|\Delta\text{RSSI}| \approx 0$ is a small change of RSSI that cannot be easily recognized by localization models. In our dataset, we find that lots of tested pairs with distance ΔGPS larger than $25m$ have almost zero change of RSSI, i.e., $|\Delta\text{RSSI}| \approx 0$. This phenomenon confirms that the sensitivity of MR data within the radius $25m$ is too low to be recognized in practice. To enhance the localization performance in telco networks, we may improve the sensitivity of sensor (wireless sector in this paper) by designing new hardware or signal measurement methods. Our experiments in Section 4 support this analysis. CCR can achieve this lower limit after providing sufficient training data.

3.3 Uncertainty

Uncertainty reflects the phenomenon that MDs do not always connect to the closest sector or BS because of multipath, non-of-sight signal propagations and multi-access interference. It is one major source of large localization errors because the RSSI is very weak if an MD connects to a farther sector or BS. Figure 5 (a) shows the MR data distribution over ΔDistance,

$$\Delta\text{Distance} = ||\text{GPS}_{\text{connecting BS}} - \text{GPS}_{\text{closest BS}}||_{\text{earth}}, \quad (3)$$

which is the distance between the connecting BS and the closest BS. If an MD connects to the closest BS, then $\Delta\text{Distance} = 0$. We see that around 50% data connect to the BS within $100m$ of the closest one. Generally, if $\Delta\text{Distance} \geq 300m$, the localization performance is not satisfactory, which occupies less than 30% of the total data. Figure 5 (b) shows the MR data distribution over the top K closest BSs. More than 50% MR data are generated by connecting with top 3 closest BSs, which often provides better localization performance. More than 30% MR data are yielded without connecting with top 6 closest BSs, which often produce worse localization performance. Our experiments in Section 4 confirms this trend. Uncertainty may be improved in near future by adding more BSs in the cell, which will assure an MD always find the closest BS for stronger RSSI values as well as better service quality.

3.4 Missing Value

Not all MR records have RSSI values from 6 nearby sectors or BSs [24]. Generally, the more values the better localization performance. Figure 6 shows the MR data distribution over the number of

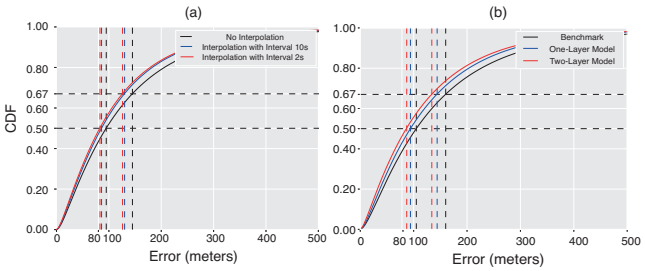

Figure 7: Comparisons of different CCR model settings: (a) Effectiveness of interpolation, and (b) Effectiveness of two-layer regression models.

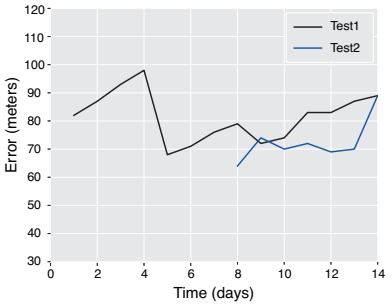

Figure 8: CCR on temporal stability of MR data.

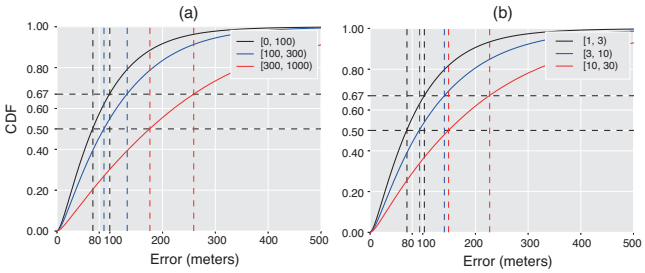

Figure 9: CCR performance on connection uncertainty of MR data: (a) ΔDistance and (b) Top K.

associated BSs (a) and sectors (b). Some MR records have RSSI values from different sectors in the same BS. This is the reason why Figure 6 (b) shows more percentage of data having 6 sectors than that of having 5 sectors. Experiments in Section 4 shows that the RSSI values from different sectors in the same BS cannot improve the localization accuracy. For example, MR data with RSSI values from 6 sectors do not produce higher localization accuracy than those with values from 5 sectors. We see that more than 60% of MR data have RSSI values from less than 2 location-associated BSs, which is the major source of large localization errors. There are two main methods to overcome missing values in MR data: 1) adding more BSs or improving sensor systems; 2) designing missing value imputation algorithm to estimate missing values. These may be our future work for further studies.

4. LOCALIZATION PERFORMANCE

The extensive experiments run on a platform consisting of six Huawei RH2288 servers with Intel(R) Xeon(R) CPU E5-2690 v2 @ 3.00GHz, 40 Cores and 189G Memory. We partition the GPS-associated MR data in Table 5 into training and test data according to the time stamp. Without specific descriptions, we use the first 24 days for training and the remaining 6 days for test purposes, where the training data take around 80% and test data take around 20%. In CCR, we use a total of 540 trees in RF distributed into 6 machines. For each block of data in Table 5, we train a CCR localization model. As a result, we train a total of 16 CCR models for the centroid of Shanghai city, China. We use the localization error, which is the earth distance between the predicted position and the ground truth GPS position, to evaluate the performance of each CCR model. We use the cumulative density function (CDF) [22, 16] of errors with respect to the proportion of the test data, where the median (50%), mean and 67% error are reported. We also use the area under curve (AUC) of CDF with the error less than $100m$ to measure the performance of CCR. The metric $\text{AUC}_{100m} \in [0, 1]$ is the proportion of test data with the error less than $100m$. The larger AUC_{100m} the better localization performance.

4.1 CCR Performance

Figure 7 compares the performance of CCR in different settings. First, in Figure 7 (a), we compare CDFs of CCR under three map matching and interpolation settings as shown in Section 2: 1) no interpolation, 2) interpolation with the 10 second time rate, and 3) interpolation with the 2 second time rate. Different interpolation time rates will lead to different training data volume. For example, the volume of training data with the 2 second time rate is around 5 times larger than that with the 10 second time rate. We see that more training data will lead to lower errors: $\text{AUC}_{100m} = \{0.2618, 0.2865, 0.2984\}$ for three settings, respectively. In the meanwhile, the median error drops from $95.7m$ (no interpolation) to $83.5m$ (2 second interpolation) and the mean error drops from $135.3m$ (no interpolation) to $118.0m$ (2 second interpolation). The map-matching and interpolation with 2 second rate gains 14% improvement over no interpolation in terms of AUC_{100m}. If we add more training data by more days (e.g., 12 days versus 29 days of training data when the same 1 day test data are used), the median error reduces from $84.7m$ to $81.1m$. Although such an improvement is not that significant (less than 5%), it also confirms the business value of big training data volume, which is consistent with the conclusion made in [13]: bigger is really better.

Second, in Figure 7 (b), we compare CCR in three conditions: 1) benchmark one-layer model without RSSI features, 2) RSSI features of one-layer model, and 3) RSSI features of two-layer model. The benchmark is proposed in [22, 16] where the RSSI values are not used as features. Instead, only feature vectors about sectors, such as ID, location and engineering parameters in Table 3 are input to one-layer RF regression model for localization. Similarly, the one-layer model uses the features in Table 3 including RSSI values for regression. The two-layer model is the proposed CCR, which uses all features in both Tables 3 and 4 to incorporate coarse-to-fine features. We see that two-layer model performs much better than one-layer model as well as benchmark. For three conditions, $\text{AUC}_{100m} = \{0.2347, 0.2647, 0.2984\}$, respectively. Two-layer model gains 12.7% and 27% improvements over one-layer model and benchmark, respectively. The median/mean error reduces from $105.8m/148.7m$ (benchmark) and $94.8m/132.5m$ (one-layer model) to $83.5m/124.3m$ (two-layer model), which confirms the effectiveness of incorporated contextual fine-grained features. This reconfirms the conclusion made in [13]: enriching the variety of features will improve the predictive performance.

We partition the 24 day training data into $4 \times 4 = 16$ blocks for 16 small CCR models. Table 6 shows the training time and memory usage of one CCR model on 6 servers, where each server is

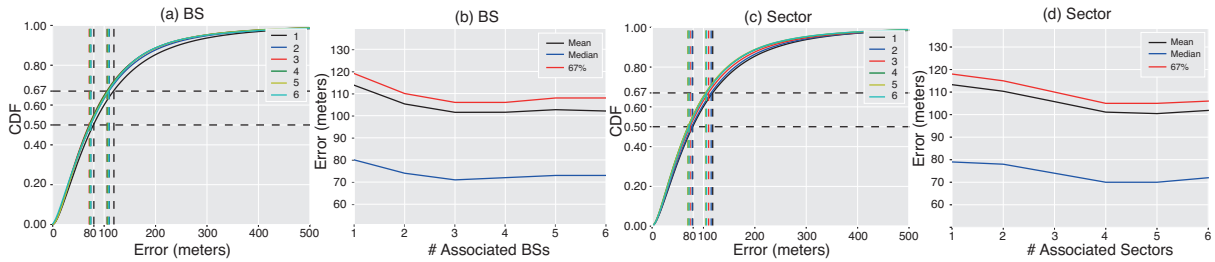

Figure 10: CCR localization performance on MR data with different missing number of BSs and sectors.

Table 6: CCR Training Time and Space.

Step	Time (minutes)
Coarse feature engineering and preparation	7.00
Training time of the first layer model	22.99
Fine-grained feature engineering	6.54
Training of the second layer model	26.65
Model	**Space (memory size)**
First-layer RF	5.7GB
Second-layer RF	5.3GB

allocated 90 regression trees for training. The coarse feature engineering and preparation requires around 7 minutes for each data block. The training time of the first layer of CCR is around 23 minutes, and output the predicted location of training samples. It takes around 6.5 minutes to compose the contextual fine-grained features. The training time of the second layer of CCR is around 27 minutes. Totally, we need around 5.7GB and 5.3GB memory to store the parameters of the first and second layer regression trees. When compared with the histogram-based fingerprinting method [14], the time and space complexities of CCR is much lower. In working scenarios (prediction phase), CCR maps the input feature vector to the predicted coordinates with a significantly faster speed than that in training phase.

4.2 CCR on Four Properties of MR

Figure 8 shows the performance of CCR on the temporal stability of MR data. Test1 uses $1 \sim 7$ days of training data and evaluates the median errors for the next $8 \sim 21$ days (test data). In Section 3.1, we have shown that the temporal stability of RSSI over days will affect the localization performance. We see that the median error curve varies from $70m$ to $100m$ in Test1, which consists with our analysis in Figure 3. Also, we find that the trained CCR performs steadily worse when the time flies, which indicates that we need to update CCR regularly based on new coming training data. To verify this hypothesis, Test2 uses new coming training data $8 \sim 14$ days and evaluates the median errors of the next $15 \sim 21$ days (test data). We find that Test2 has the overall median error lower than that in Test1, which confirms that updating CCR regularly based on new coming data really captures the recent temporal variance in MR data. Another important observation is that weather also influences the localization performance. Raining or bad weather often degrades the localization performance in Test1, where days $\{2, 3, 4, 2, 13, 14\}$ are big raining days with an average median error $89.5m$, while other days have an average median error $75.6m$ in Test1. This is our future work to enhance the robustness of CCR to different weather conditions.

Generally, CCR produces a median error of $80m$ in Figure 7, which still has a gap with the lower limit of the median error $50m$ based on the spatial sensitivity of MR data in Figure 4. One possible reason is that the training data is still insufficient for CCR to achieve this lower limit. To verify this hypothesis, we collect more training data (52998 samples) by fine-grained walk with $0.5m$ per step around sectors A and B in Figure 4. On the test data (46917 samples), we obtain a median error around $46.1m$, which is almost the same with the lower limit $50m$. Future work may focus on increasing sensitivity of MR data either by designing new sensors or by new algorithms for better localization performance.

Figure 9 evaluates the performance of CCR on uncertainty of MR data. In (a), we see that the bigger ΔDistance defined in Eq. (2) has the larger errors, consistent with our analysis in Figure 5. For example, $\Delta\text{Distance} \in [0, 100)$ has a median error $68m$ and a mean error $88.8m$, while $\Delta\text{Distance} \in [300, 1000)$ has a median error $176m$ and a mean error $223.7m$. Obviously, we can see smaller ΔDistance has smaller locating error. This is because the closer connecting sectors provide stronger and more stable RSSI values. So the locating error is reduced with the decrease of ΔDistance. In Figure 9 (b), we see that connecting top $[1, 3)$ closest sectors have a median error $70m$ and a mean error $92m$, while connecting top $[10, 30)$ closest sectors have a median error $149m$ and a mean error $199m$. When K increases, the error also increases. These results are consistent with the observations in Figure 9 (a). Through extensive empirical studies, we find an approximate rule on localization errors:

- The average localization error is often within $1/4 \sim 1/2$ of the distance between the MD and the connecting BS.

As a result, it is necessary to reduce the uncertainty of MR data, i.e., force the MD connecting the closest sector or BS. In near future, LTE may build more BSs to reduce the cell size in cellular network, which is a method to reduce the uncertainty of MR data because non-line-of-sight signal propagation rarely happens.

Figure 10 shows the performance of CCR on MR data with different number of missing values. Generally, less missing values lead to higher localization accuracy. In (a) and (b), MR data with 3 associated BSs produce the median error $71m$ and the mean error $101m$, while MR data with 1 associated BS get the median error $80m$ and the mean error $113m$. We find that more associated BSs will not provide higher localization accuracy. For example, MR data with 4 associated BSs have the median error $72m$ and the mean error $101m$, almost the same with those for MR data with 3 associated BSs. The same phenomenon occurs in (c) and (d) for missing sectors. Note that MR data with RSSI values from 6 sectors do not produce higher localization accuracy than those with values from 5 sectors. This is because some RSSI values are from different sectors in the same BS. Therefore, MR data with 3 associated BSs are enough to give satisfactory localization accuracy. However, Figure 6 shows that more than 50% MR data have RSSI values from less than 3 BSs. This result suggests us design specific missing value imputation algorithms or use more contextual information of sequential trajectory point to rectify localization errors of MR data with 1 or 2 associated BSs.

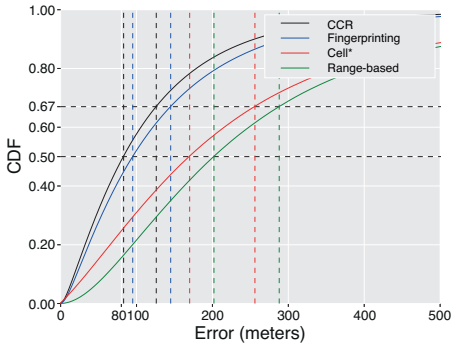

Figure 11: Comparison of CCR with the state-of-the-art localization methods: range-based, Cell* and fingerprinting methods. CCR performs the best with a median error 80m.

4.3 CCR Comparisons and Error Analysis

Figure 11 compares CDF of CCR with those of the state-of-the-art localization methods: 1) the range-based method [24], 2) the fingerprinting method [14], and 3) Cell* [16]. We use the same training and test sets to evaluate different models. We see that the range-based method performs the worst because of the complex radio signal propagation patterns in urban areas. The range-based method has $\text{AUC}_{100m} = 0.082$, a median error $202m$ and a mean error $308m$, which is consistent with the results in [24]. Cell* works better than the range-based method since some contextual knowledge are used through map-matching with the sector switching sequence. However, Cell* does not use RSSI features so that it has slightly lower localization accuracy than the fingerprinting method. Cell* has $\text{AUC}_{100m} = 0.1496$, a median error $157m$ and a mean error $263m$. The fingerprinting method performs much better than Cell* with $\text{AUC}_{100m} = 0.2618$, a median error $95.7m$ and a mean error $135.3m$. The proposed CCR performs the best because it designs a context-aware and coarse-to-fine features, which can capture much more signal and context knowledge than previous methods. Using AUC_{100m} metric, CCR outperforms $\{264\%, 99.5\%, 14\%\}$ when compared with the range-based, Cell* and fingerprinting methods.

Figure 12 shows some typical localization errors of CCR, where the blue bubbles are the predicted locations and the other end of the red line is the ground-truth GPS location. Figure 12 (a) shows that the predicted locations have small errors on the same road. These are the expected results by using knowledge of road networks and fine-grained contextual features. Figure 12 (b) shows location points correctly predicted on the same road but with forward or backward errors along the road. This type of error happens in most cases because training data are insufficient to cover fine-grained location. The locations close to the ground truth position of the test data are missing in the training data. So, the predicted position is approximated by nearby locations in the training data with large errors. The solution is to collect more fine-grained training data for CCR learning. Figure 12 (c) shows some predicted locations along the close and parallel road of the ground truth road. This type of error is often caused by the low temporal stability and spatial sensitivity of MR data, which cannot be also rectified by the road network knowledge or contextual features because of the parallel structure. This indicates that in the future we should collect denser training data and design special algorithm in the areas with close parallel roads than other areas. Finally, Figure 12 (d) shows that large errors occur in dense road networks. This also indicates that we should collect more training data on the areas with dense road networks than normal areas to distinguish subtle details among dense roads. Alternatively, we should also consider using larger time window to capture long-term contextual knowledge.

5. RELATED WORK

Localization techniques in telco network can be broadly divided into two categories: 1) range-based methods and 2) fingerprinting methods. The range-based methods are defined by protocols that use absolute point-to-point distance estimates or angle estimates for calculating location [4]. Usually, they extensively use physical models of radio signal propagation and combinations of MR features such as RSSI, TOA/TDOA, and AOA for range estimation [9, 21, 6, 10, 11]. The solutions in range-based localization in telco networks generally have two main weaknesses. First, the signal measurements are often noisy and influenced by multipath propagation, non-line-of-sight propagation and multiple access interference [4]. Second, some MR data are unavailable in real-world telco networks due to efficiency problems.

Fingerprinting methods are more accurate than range-based localization strategies [22, 14, 16]. A fingerprint database stores the mapping function from RSSI feature vector to the corresponding ground truth GPS coordinates. This is often constructed once in an offline phase. Online localization prediction is performed by querying the mapping coordinates given RSSI feature vector without location labels. Fingerprinting methods often require lots of training data (i.e., RSSI features with GPS labels) to learn the accurate fingerprint. As the sate-of-the-art fingerprinting benchmark in this paper, CellSense [14] divides the area of interest into small grids, and constructs the RSSI vector histogram as the probabilistic fingerprint for each grid. When locating an MD given the RSSI vector, CellSense searches its K nearest neighbors in the fingerprint database, and returns the weighted average position of the neighbors. The grid size is an adjustable parameter to balance the scalability and accuracy of the fingerprint. If the training data are sufficiently large by dense GPS labels, the fingerprint can find fine-grained GPS coordinates given RSSI vectors. The average error of fingerprinting methods is around $100 \sim 200m$. One challenge is that the time cost of building a city-scale fingerprint is often high [14]. Note that some fingerprinting methods [22, 16] do not use RSSI vectors. Instead, they use only the GPS-associated BS or sector switching sequence to learn a sequential fingerprint. As the state-of-the-art benchmark called Cell* [16], a BS or sector switching sequence (trajectory) of an MD is mapped to a sequence of grids with GPS coordinates according to road network constraints. Cell* achieves a median error of $230m$ for the stationary location estimation and a median error of $70m$ for mobility path estimation.

However, most of fingerprinting methods suffer from insufficient training data to build the fingerprint database. Normally, there are two main methods to obtain the big GPS-associated MR data. First, cars drive [22, 14] or people walk [16] in road networks with GPS equipments to collect GPS-associated MR data. Obviously, this method is cost-consuming by hiring cars and people for the city-scale area coverage. Moreover, updating fingerprint database requires to collect new coming training data periodically, which increases the overall costs. Second, turning on AGPS (assisted global positioning system) from telco networks can obtain GPS-associated MR data of most MDs. However, AGPS will cause serious extra energy consuming of MDs for bad user experiences. In practice, it is impossible to build and maintain the city-scale fingerprint database without new coming training data.

Different from the above solutions, we deploy a cost-efficient regression model called CCR in telco big data platform: 1) Training data are obtained by integrating MR data with LBS OTT GPS data

(a) (b) (c) (d)

Figure 12: Localization error analysis (blue bubble is the predicted location and the other end of the red line is the ground-truth GPS location): (a) small errors on the same road. (b) large errors on the same road. (c) large errors on the parallel road. (d) large errors on dense road networks.

from telco big data platform. 2) Training random forest-based CCR is time-efficient on distributed Hadoop/Spark systems. 3) Map-matching and interpolation are used to capture contextual information of road networks. 4) CCR has a cascaded architecture to encode coarse-to-fine contextual features.

6. CONCLUSIONS

In this paper, we describe a novel city-scale localization model with telco big data called CCR, which has been deployed in Spark and Hadoop-based telco big data platform to provide location insights of customer movement behaviors. Through automatic training data acquisition, CCR uses the map-matching and interpolation algorithms to obtain big GPS-associated MR data, which encodes contextual information of road networks. Also, CCR adopts a two-layer RF-based regression model to capture coarse-to-fine contextual location features. The performance of CCR is superior to the state-of-the-art range-based and fingerprinting methods. Extensive measurements and experiments are carried out to obtain practical implications and guidelines for improving localization accuracy in telco networks: 1) Collecting more fine-grained training data. 2) Update models frequently to capture the variance of MR data. 3) Encoding more structural information of dense road networks. 4) Building more BSs to reduce cell size of cellular networks. 5) Enhancing temporal stability and spatial sensitivity. 6) Reducing uncertainty and missing values in MR data.

Acknowledgment

This work was supported by NSFC (Grant No. 61373092, 61572365, 61272449 and 61572339). This work was partially supported by Collaborative Innovation Center of Novel Software Technology and Industrialization, and Science and Technology Commission of Shanghai Municipality (Grant No. 15ZR1443000).

7. REFERENCES

[1] H. Aly and M. Youssef. Dejavu: An accurate energy-efficient outdoor localization system. In *SIGSPATIAL*, pages 154–163, 2013.

[2] F. Bandiera, A. Coluccia, and G. Ricci. A cognitive algorithm for received signal strength based localization. *IEEE Transactions on signal processing*, 63(7):1726–1736, Apr. 2015.

[3] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[4] J. Caffery and G. Stuber. Overview of radiolocation in CDMA cellular systems. *IEEE Communications Magazine*, 36(4):38–45, 1998.

[5] F. Calabrese, L. Ferrari, and V. D. Blondel. Urban sensing using mobile phone network data: A survey of research. *ACM Computing Surveys*, 47(2):25, 2015.

[6] A. Catovic and Z. Sahinoglu. The Cramer-Rao bounds of bybrid TOA/RSS and TDOA/RSS location estimation schemes. *IEEE Communications Letters*, 8:626–628, 2004.

[7] M. Y. Chen, T. Sohn, D. Chmelev, D. Haehnel, J. Hightower, J. Hughes, A. Lamarca, F. Potter, I. Smith, and A. Varshavsky. Practical metropolitan-scale positioning for GSM phones. In *UbiComp*, pages 225–242, 2006.

[8] Y.-C. Cheng, Y. Chawathe, A. LaMarca, and J. Krumm. Accuracy characterization for metropolitan-scale Wi-Fi localization. In *MobiSys*, pages 233–245, 2005.

[9] L. Cong and W. Zhuang. Hybrid TOA/AOA mobile user location for wideband CDMA cellular systems. *IEEE Transactions on Wireless Communication*, 1(3):439–447.

[10] S. Gezici. A survey on wireless position estimation. *Wirel. Pers. Commun.*, 44(3):263–282, 2008.

[11] S. Hara, D. Anzai, T. Yabu, T. Derham, and R. Zemek. Analysis on toa and tdoa location estimation performance in a cellular system. In *IEEE International Conference on Communications*, pages 1–5, 2011.

[12] X. Hu, M. Yuan, J. Yao, Y. Deng, L. Chen, Q. Yang, H. Guan, and J. Zeng. Differential privacy in telco big data platform. In *VLDB*, pages 1692–1703, 2015.

[13] Y. Huang, F. Zhu, M. Yuan, K. Deng, Y. Li, B. Ni, W. Dai, Q. Yang, and J. Zeng. Telco churn prediction with big data. In *SIGMOD*, pages 607–618, 2015.

[14] M. Ibrahim and M. Youssef. Cellsense: An accurate energy-efficient GSM positioning system. *IEEE Transactions on Vehicular Technology*, 61(1):286–296, 2012.

[15] S. Jiang, G. A. Fiore, Y. Yang, J. Ferreira Jr, E. Frazzoli, and M. C. González. A review of urban computing for mobile phone traces: current methods, challenges and opportunities. In *KDD Workshop on Urban Computing*, pages 2–9, 2013.

[16] I. Leontiadis, A. Lima, H. Kwak, R. Stanojevic, D. Wetherall, and K. Papagiannaki. From cells to streets: Estimating mobile paths with cellular-side data. In *ACM CoNext*, pages 121–132, 2014.

[17] Y. Li, C. Chow, K. Deng, M. Yuan, J. Zeng, J. Zhang, Q. Yang, and Z. Zhang. Sampling big trajectory data. In *CIKM*, pages 941–950, 2015.

[18] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang. Map-matching for low-sampling-rate GPS trajectories. In *ACM SIGSPATIAL GIS*, pages 352–361, 2009.

[19] C. Luo, J. Zeng, M. Yuan, W. Dai, and Q. Yang. Telco user activity level prediction with massive mobile broadband data. *ACM Transactions on Intelligent Systems and Technology*, 2016.

[20] W. Luo, H. Tan, L. Chen, and L. M. Ni. Finding time period-based most frequent path in big trajectory data. In *SIGMOD*, pages 713–724, 2013.

[21] D. Niculescu and B. Nath. Ad hoc positioning system (APS) using AOA. In *INFORCOM*, pages 1734–1743, 2003.

[22] J. Paek, K.-H. Kim, J. P. Singh, and R. Govindan. Energy-efficient positioning for smartphones using Cell-ID sequence matching. In *ACM MobiSys*, pages 293–306, 2011.

[23] M. Yuan, K. Deng, J. Zeng, Y. Li, B. Ni, X. He, F. Wang, W. Dai, and Q. Yang. OceanST: A distributed analytic system for large-scale spatiotemporal mobile broadband data. In *VLDB*, pages 1561–1564, 2014.

[24] H. Zang, F. Baccelli, and J. Bolot. Bayesian inference for localization in cellular networks. In *INFORCOM*, pages 1–9, 2010.