Experience: Large-scale Cellular Localization for Pickup Position Recommendation at Black-hole

Shuli Zhu[1], Lingkun Li[1], Xuyu Wang[2], Changcheng Liu[3], Yuqin Jiang[3],
Zengwei Huo[3], Hua Chai[3], Jiqiang Liu[1], Dan Tao[1], Ruipeng Gao[1*]

[1]Beijing Jiaotong University, [2]Florida International University, [3]DiDi

[1]{zhushuli, lkli, jqliu, dtao, rpgao}@bjtu.edu.cn, [2]xuywang@fiu.edu

[3]{corlosliu, jiangyuqin, huozengwei, chaihua}@didiglobal.com

[*]Corresponding author

ABSTRACT

Location awareness is the basis for enabling pickup service at ride-hailing platforms. In contrast to the almost pervasive coverage outdoors, indoor localization availability is still sporadic in industry since it largely relies on RF signatures from certain IT infrastructure, e.g., WiFi access points. Based on our 2-year observations at DiDi ride-hailing platform in China, there are $68k$ orders everyday created at black-hole, i.e., where only cellular signals exist. In this paper, we present the design, development, and deployment of *TransparentLoc*, a large-scale cellular localization system for pickup position recommendation, and share our 2-year experience with 50 million orders across 13 million devices in 4541 cities to address practical challenges including sparse cell towers, unbalanced user fingerprints, and temporal variations. Our system outperforms the iOS built-in cellular localization system in terms of four major service metrics, regardless of environmental changes, smartphone brands/models, time, and cellular providers.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; • **Information systems** → **Location based services**; • **Applied computing** → *E-commerce infrastructure.*

KEYWORDS

Cellular Localization, Pickup Position Recommendation, Ride-hailing Platform, Mobile Crowdsensing

1 INTRODUCTION

Indoor localization techniques have evolved in the past decade alongside the growth of mobile networks, enabling mobile applications such as indoor navigation and rescue services to offer fine-grained, high-quality user positioning. While Global Positioning System (GPS) is commonly used, research efforts have explored alternatives like WiFi, Bluetooth, ultrasound, and visible light due to line-of-sight (LOS) constraints with satellites [5]. However, many locations lack the necessary infrastructure, such as WiFi routers or ultrasonic speakers, to provide indoor localization services, such as underground parking lots at airports or subway stations.

In this paper, we share our two-year experience providing accurate pickup services to passengers on DiDi, a prominent ride-hailing platform akin to Uber and Lyft. The fundamental service offered by ride-hailing applications is to connect drivers with passengers, ensuring that the driver arrives at the exact location where the passenger is waiting. This crucial process relies on a pickup service that utilizes the passenger's position to recommend an optimal pickup location.

Our specific focus lies in addressing the challenges posed by indoor environments known as "black-holes." These locations lack access to reliable positioning technologies such as GPS signals, pre-collected WiFi fingerprint data, and dedicated hardware deployment. As a result, users in these areas heavily rely solely on cellular signals for their location awareness. A remarkable observation is that over 68,000 daily travel orders, accounting for approximately 2% of the

total, originate from these black-hole locations on the DiDi platform. This emphasizes the significance of our efforts in providing efficient pickup services to users in such challenging areas.

To accomplish this, the pickup position recommendation system relies on estimating the passenger's location from cellular measurements, which proves to be a challenging task. The basic cellular positioning method, Cell ID (CID [34]), suffers from significant location errors, often in the range of hundreds of meters, due to the wide coverage of cell towers. Margolies *et al.* [22] leverage cellular signatures from multiple cell towers to mitigate environmental interference. Notably, DeepLoc [31] pioneers the use of neighboring cell towers in training a deep learning model with geo-tagged 4G cellular signatures as fingerprints, resulting in more accurate and efficient user positioning compared to GPS.

Despite these advancements, large-scale deployment in industry, particularly with the emergence of the 5G NR network, poses challenges. Cell towers' limited coverage and penetrability compared to network connectivity [41], ongoing 5G infrastructure construction, and insufficient user fingerprints near black-holes all impact localization robustness. Furthermore, severe variations in cellular signatures, especially with 5G, present additional obstacles to accurate localization.

We developed a ubiquitous cellular localization system called *TransparentLoc*, which leverages existing User Measurement Data (UMD) for large-scale deployment without the need for special hardware or fine-grained indoor fingerprint collection. Instead, *TransparentLoc* uses large-scale outdoor trajectories with geo-tags to incrementally construct the cellular fingerprint set through crowdsensing, providing indoor/outdoor location inference even in GPS-denied "black-hole" areas.

To achieve accurate and scalable cellular localization at a large scale in the industry, we addressed the following challenges. First, we tackled the unpredictable temporal-spatial variations in cellular signatures by proposing a cell tower augmentation mechanism, which enhances signature dimensions and periodically updates fingerprints to combat environmental interference, device diversity, and sparse cell towers.

Second, to address the substantial storage and computation costs of constructing country-level fingerprint sets, we introduced an effective feature extraction and lightweight storage mechanism for processing incremental crowd-sourced data.

Finally, we overcame the difficulty of precisely learning the arbitrary distribution of cellular signatures and practical correlations among different cell towers by adopting deep neural models with real-time requirements. Specifically, we used a meticulous Convolutional Neural Network (CNN) model for cellular localization and a DeepFM model for pickup position recommendation.

Our large-scale experiments, based on approximately 50 million ride-hailing orders across 13 million devices, demonstrate the superiority of our cellular localization system. The recommended pickup positions generated by *TransparentLoc* achieve a 4.58% lower distance error compared to the iOS built-in cellular localization system in the median. Additionally, our system outperforms the iOS-based system in various service metrics, irrespective of environmental changes, smartphone brands/models, time, and cellular providers.

Our contributions are listed as follows:

- **Practical Deployment and Evaluation:** We present the culmination of two years of extensive research, where we have designed, deployed, and evaluated a large-scale pickup location recommendation system. This system encompasses a vast dataset of 50 million travel orders across 13 million devices, spanning 4,541 cities. By sharing our practical experiences, we aim to provide valuable insights into the real-world implementation of such systems.

- **Novel Crowdsensing Approach:** We have developed and implemented a pioneering crowdsensing approach that eliminates the need for labor-intensive indoor fingerprint collection. It overcomes common challenges encountered in real-world scenarios, such as sparse tower coverage, unbalanced fingerprints, and long-term variations. By employing this approach, we address the limitations of existing techniques and offer an improved solution for accurate cellular localization.

- **New Service Metrics for Large-Scale Evaluation:** To ensure the comprehensive assessment of our system's performance in diverse environments, we explore new service metrics specifically designed for large-scale evaluation. These metrics alleviate the need for dedicated manual efforts to measure the ground truth. By leveraging these novel metrics, we demonstrate the robustness and reliability of our system, providing a more accurate evaluation of its effectiveness.

2 CELL TOWER AUGMENTATION

2.1 Challenges in Cellular Localization

As shown in Figure 1, current smartphones can report User Measurement Data (UMD) in 4G LTE networks, which includes wireless channel measurements like Reference Signal Received Power (RSRP), Received Signal Strength Indicator (RSSI), and Reference Signal Received Quality (RSRQ)[15][1].

---

[1]In 5G NR network, RSRP and RSRQ slightly change to Synchronization Signal Reference Signal Received power (SS-RSRP) and Secondary Synchronization Signal Reference Signal Received Quality (SS-RSRQ), respectively. We still use RSRP and RSRQ in this paper for consistency.

(a) 4G LTE   (b) 5G NR
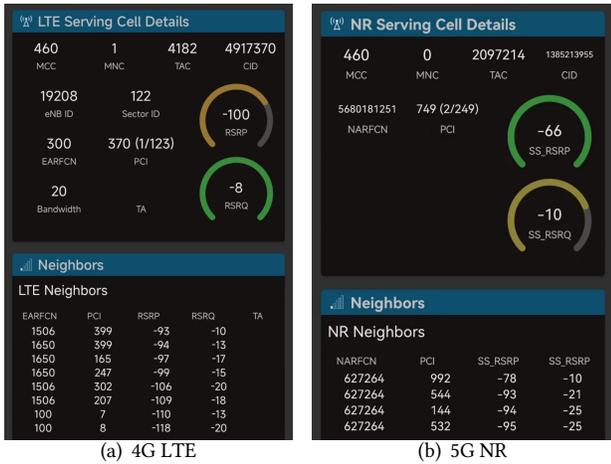
Figure 1: User Measurement Data (UMD) in 4G LTE and 5G NR network, recorded by a commodity app "Network Survey [2]".

Table 1: Proportion of the number of scanning cell towers in both 4G and 5G networks

|	1 Tower	2 Towers	3 Towers	>3 Towers
4G LTE	44.82%	4.83%	7.68%	42.67%
5G NR	53.48%	2.68%	5.50%	38.34%

RSRP represents the linear average of the received signal power of resource elements and is an important power indicator of the mobile network. A strong RSRP reading generally indicates better channel quality, suggesting proximity to the cell station. RSSI is the average value of all signals, such as pilot signals, data signals, neighboring interference signals, and noise signals, and it can be used to compute the quality of SS-RSRP measurements. RSRQ is the ratio of RSRP and RSSI.

Although these channel measurement values theoretically enable distance estimation to and from cell stations based on wireless signal propagation models, complex wireless channel environments, such as occlusions in urban cities, often result in significant localization errors.

In addition to measuring connection states to the primarily connected tower, mobile devices also record readings from **neighbouring towers**, enriching cellular signatures for positioning. This technique has been adopted in state-of-the-art academic research [22]. Intuitively, distances to neighboring cell towers help narrow down the user's possible area, enhancing localization accuracy.

However, deploying a large-scale cellular localization system at the city level presents challenges due to the sparse deployment of cell towers, especially for the rising 5G NR network, currently under construction in many countries[2].

---

[2]Most smartphones do not enable 4G LTE and 5G NR network connectivity simultaneously.

(a) Cell towers   (b) User fingerprints (8 hours)
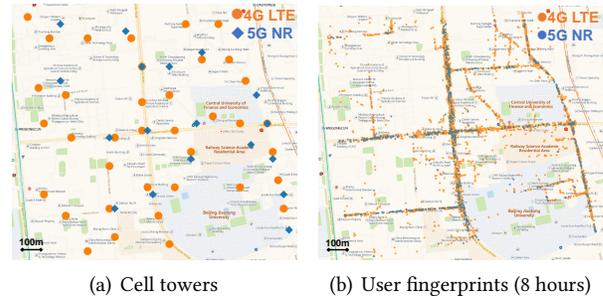
Figure 2: Deployment of 4G LTE and 5G NR networks in the central region ($2km \times 2km$) in Beijing. The number of 4G cell towers are more than 2x of 5G (4G: 37, 5G: 18), and 4G user fingerprints are more than 5x of 5G (4G: 3550, 5G: 668).

(a) RSRP in 4G LTE   (b) RSRP in 5G NR
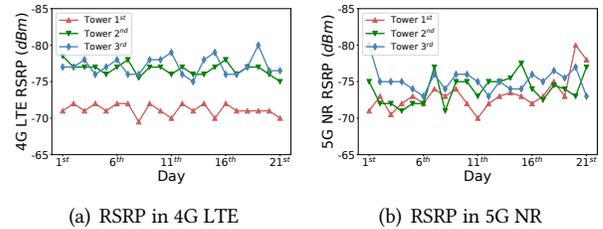
Figure 3: Temporal variations on cellular signatures from the top 3 strongest cell towers over three weeks.

This hinders the accuracy and robustness of cellular localization, limiting its usage mainly to emergency rescues. Below, we present observations from a $2km \times 2km$ test region over a month and highlight three typical challenges in deploying cellular localization in the industry.

(1) Sparse cell towers. Cell towers are sparsely deployed to efficiently cover large areas, and there are significantly fewer 5G towers compared to 4G since the 5G infrastructure is still in development. Figure 2(a) illustrates that there are more than a double of 4G cell towers than 5G even in the central region in Beijing. Furthermore, 5G towers have short-range coverage (e.g., $100 \sim 300m$) and weak penetrability, resulting in infrequent 5G signatures from neighboring towers. Based on our analysis with 1 million travel orders, Table 1 shows that almost half of the orders can only detect the connected cell towers. Notably, 5G mobile users generally hear fewer cell towers compared to 4G users.

(2) Unbalanced user fingerprints. The current quantity of 4G users far exceeds that of 5G users due to mature 4G deployment and cost-effective 4G smartphones. Figure 2(b) shows that there are over 5 times more 4G user fingerprints than 5G in the same test region. This imbalance hampers the robustness of 5G localization across diverse smartphones, locations, and orientations.

(3) Temporal variations. Figure 3 depicts daily RSRP signatures of the top 3 strongest cell towers in 4G and 5G

networks at a random location from Figure 2 over three weeks. Since the measured RSRP values vary all in the same region, capturing long-term dependency and resisting frequent temporal variations are necessary for reliable cellular localization. In addition, our dataset over China suggests that 4G signatures exhibit slightly stable compared to 5G, e.g., the RSRP measured in 5G was observed to be $-77 \pm 11.4dBm$, while it was measured as $-83 \pm 11.05dBm$ in 4G.

2.2 Augmented Cell Tower as a Primitive

We aim to identify more available neighbouring cell towers via existing mobile Application Program Interface (API), i.e., leveraging only public measurements on commodity smartphones for ubiquitous cellular positioning. We find that other than the Cell ID (i.e., CID), there are more stable and useful information such as PCI and EARFCN in 4G LTE network (NR-ARFCN in 5G NR network) [15]. Specially, PCI stands for Physical Cell ID, which represents both Group Cell ID and Section Index of a 4G base station, i.e.,

$$N_{PCI} = 3N_{Cell} + N_{Sec} \tag{1}$$

where $N_{Cell} \in [0, 167]$ is the Group Cell ID, $N_{Sec} \in \{0, 1, 2\}$ is the Section Index in each tower, and thus the PCI number N_{PCI} ranges from 0 to 503.

In addition, the EARFCN in 4G, which is similar to NR-ARFCN in 5G, denotes the Absolute Radio Frequency Channel Number [15]. It is a 16 bit integer ranging from 0 to 65535, representing the different frequency bands of each cell tower for communication.

Thus, we adopt (PCI,EARFCN) as the unique index for "augmented" cell towers. Specially, two augmented cell towers may locate at the same location, but orient to different directions (i.e., different sections of the tower), or transmit data at different bands. For example, there are 39 "augmented" 5G cell towers compared with only 18 real 5G cell towers over the example region in Figure 2(a). This indicates the effectiveness to enrich the dimension of cellular signatures with more heard cell towers. In addition, Android does not ensure the Cell ID information for neighbouring cell towers, thus our mechanism also increases the stability of discovering neighbouring cell towers in cellular localization.

3 SYSTEM OVERVIEW

Based on the augmented cell towers which provides sufficient network connectivity, we design the *TransparentLoc* system, which utilizes cellular measurements to ensure pickup point recommendation service for passengers staying at "black-holes", e.g., office buildings, shopping malls, subway stations, and underground parking structures. This service is crucial for ride-hailing platforms since passengers may issue travelling orders at anytime and anywhere.

Figure 4 shows the design overview of our system, which consists of three major phases: *fingerprint set construction* at
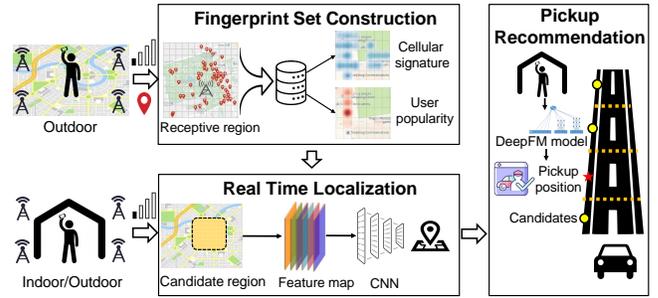
Figure 4: *TransparentLoc* **design at DiDi ride-hailing platform. It leverages outdoor trajectories with geotags to construct the cellular fingerprint set by crowdsourcing, and localizes indoor/outdoor passengers by only cellular measurements. It further recommends a customized pickup position and informs both passengers and drivers.**

outdoor with geo-tags, *localization with CNN* at indoor/outdoor by mere one-shot cellular measurements, and *pick-up point recommendation* to guide indoor passengers where to take the car.

In fingerprint set construction phase, *TransparentLoc* partitions the city map into equal grid cells, and produces the receptive region[3] for each cell tower by crowdsensed measurements. Meanwhile, cellular signatures and user popularity on each grid cell are incrementally updated and effectively stored in the fingerprint set, with resilient representations to combat arbitrary noise distribution in practice.

In localization phase, *TransparentLoc* produces an appropriate candidate region by cellular measurements from the passenger's mobile phone, and extracts a multi-dimensional feature map over the corresponding area. The joint features are fed into a meticulous CNN model to pinpoint user's relative position toward the region's center. Note that *geo-tags are only used for model training* rather than location inference.

In pick-up recommendation phase, *TransparentLoc* provides a proper pick-up point for passengers, based on their estimated locations by cellular measurements when staying at black-holes. Different from traditional recommendation methods, we have explored a specific design with three stages during its large-scale deployment, including pairwise sorting instead of binary classification, customized recommendation with personal history, and a DeepFM model for feature extraction and learning.

4 FINGERPRINT SET CONSTRUCTION

The generalizability of the fingerprint set is crucial for industry-deployed learning-based cellular localization. It should cover

---

[3]The "receptive region" refers to the geographical coverage area of a cell tower where it provides reliable cellular signal coverage to mobile devices.
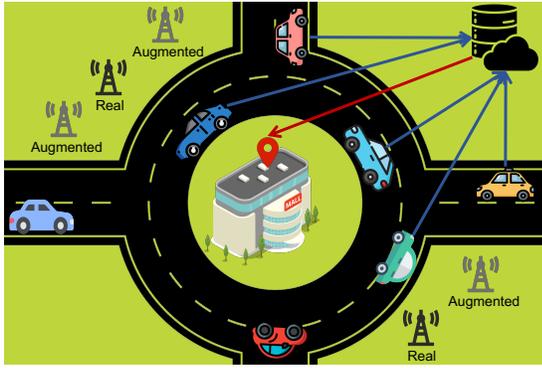
Figure 5: Fingerprint set collection process.

(a) Map grid partitioning

(b) Receptive region production

(c) Cellular signature representation
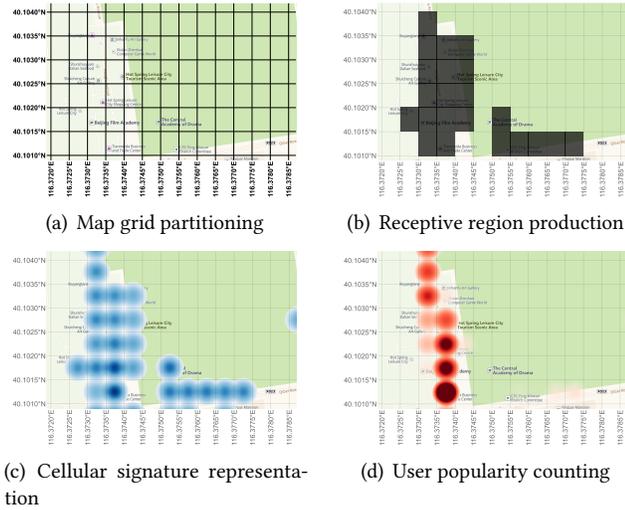
(d) User popularity counting

Figure 6: Fingerprint set construction with four steps: map grid partitioning, receptive region production, cellular signature representation, and user popularity counting.

(a) RSRP distribution

(b) RSRQ distribution

Figure 7: Arbitrary distribution of RSRP and RSRQ signatures heard from a 5G base station at a grid cell.

features, enabling us to locate users anywhere, even with fast-moving vehicles contributing to our fingerprint collection.

Receptive region production. We observe that user measurement data includes cellular signatures from both primarily-connected and neighboring cell towers. We aggregate samples with geo-tags for each augmented cell tower based on its unique index (Section 2.2) to calculate its receptive region at the grid level (Figure 6(b)). We further adopt the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm [29] to eliminate outlier grids.

Cellular signature representation. The abundance of cellular signatures from both primarily-connected and neighboring cell towers is essential for cellular positioning (Figure 6(c)). Storing such vast data as-is incurs redundant storage and heavy computation. Existing approaches, like NBL [22], assume Gaussian distribution for each cell tower, computing mean and standard deviation values for feature representation. However, this assumption doesn't hold in large-scale deployments where irregular distributions of RSRP and RSRQ are common (Figure 7). To address this, we explore a bucket-based storing mechanism [17].

After profiling all historical RSRP and RSRQ signatures in China (Figure 8), we divide the signature range into seven intervals with equal sample quantities. Each cell tower has specific buckets to store the quantity of corresponding samples based on common bucket boundaries in signatures. When any bucket value exceeds 256, all values are halved, and we repeat this operation until they are all below 256. Each bucket value is stored with one byte (e.g., Byte 1 ~ 7 in Figure 9). Additionally, another byte (e.g., Byte 0) stores the number of halving operations. Thus, we store cellular signatures with arbitrary distributions as an INT64 integer for each grid cell.

To capture the latest cellular signatures, we maintain a sliding time window of 30 days, updating bucket values for each cell tower daily.

User popularity counting. People are more likely to stay at regional hot spots, i.e., Points of Interests (POIs) such as restaurants, stores and entrances (Figure 6(d)). In order to acquire the number of active popularity at each grid cell, we employ two indicators which have been widely adopted in website access statistics [3], i.e., Page View (PV) and Unique

long periods, wide areas, and various devices to ensure accurate and robust ubiquitous localization.

At DiDi ride-hailing platform, we have constructed a country-level fingerprint set in 4,541 large/median/small cities in China over two years. Figure 5 illustrates the data collection process, leveraging crowdsensing to gather trajectories from mobile users, including accurate GPS locations (geo-tags) and associated cellular information, comprising servicing and neighboring base stations.

Figure 6 outlines the four-step construction process of our fingerprint set: map grid partitioning, receptive region production, cellular signature representation, and user popularity counting. We also employ a resilient and incremental feature representation mechanism and efficiently store this vast amount of data on the cloud.

Map grid partitioning. To gather cellular fingerprints from crowdsourced travel trajectories with geo-tags, we virtually divide the city map into equal grids (Figure 6(a)). Samples within each grid cell are aggregated to produce specific
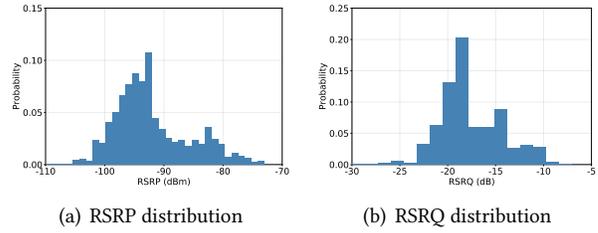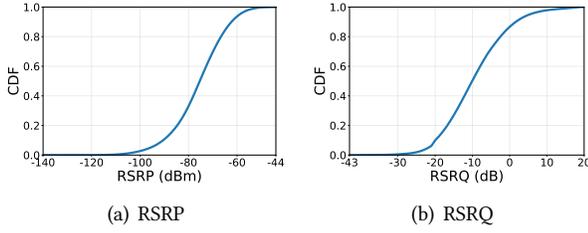
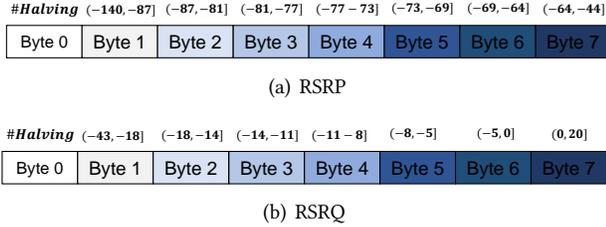Figure 8: RSRP and RSRQ signatures of all 5G cell towers over China.

Figure 9: Bucket-based storing mechanism on cellular signatures.

Figure 10: User popularity in a shopping mall along four weeks in December.

Visitor (UV). Specifically, in terms of user measurement data, we define PV as the total number of samples collected at a grid cell over some time, and UV as the number of associated users collecting the samples at this grid over the period. Since recent visits are more valuable than earlier ones, we incrementally update the counts with a Gaussian time decay factor, i.e.,

$$h_j^{PV} = \sum_{d=0}^{D} N_{j,t-d}^{PV} \cdot g(d), h_j^{UV} = \sum_{d=0}^{D} N_{j,t-d}^{UV} \cdot g(d) \quad (2)$$

$$g(d) = e^{-\frac{d^2}{2\sigma^2}} \quad (3)$$

where $N_{j,t}^{PV}$ and $N_{j,t}^{PV}$ represent PV and UV values on cell grid j at time t, respectively. D indicates the date length for statistic (30 days in our system), and σ is the Gaussian time decay factor (1.33 in our system). Figure 10 depicts the thermal maps on user popularity in a shopping mall over four weeks in December, and there are more passengers in the last two weeks due to the Christmas and New Year Festival.

In sum, our fingerprint set is stored with a tree structure in database (Figure 11). The key for each "augmented" neighbouring cell tower is the unique index (elaborated in Section 2.2), and followed by the grid Cell ID on the map. At
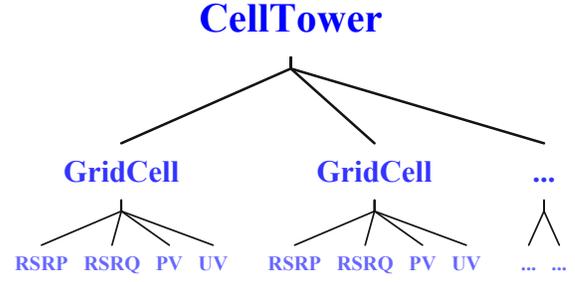

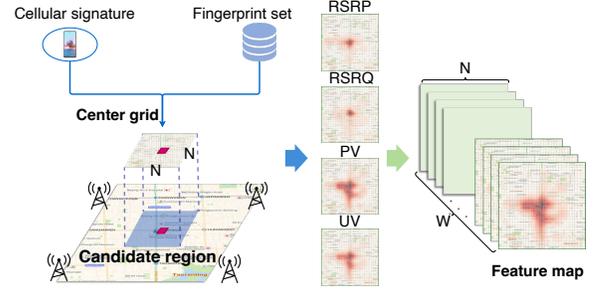Figure 11: Data format in fingerprint database.

Figure 12: Candidate region production and feature map generation.

each grid cell, its features (RSRP, RSRQ, PV and UV) are all incrementally updated everyday.

5 REAL TIME LOCALIZATION

In this section, we measure the real-time UMD from both indoor/outdoor mobile users, and localize them in real time based on our large-scale fingerprint set. As shown in Figure 12, our system consists of candidate region production, feature map generation, and CNN model training/inference.

5.1 Candidate Region Production

Estimating an appropriate candidate region is crucial and efficient in cellular positioning, e.g., determining whether the user is inside a shopping mall or at a restaurant without searching all fingerprints. The main challenge involves the precision of central grid cell and the proper size of region area (evaluated in Section 7.8). Too small regions may miss correct locations outside the region, while too wide regions cause a larger number of model parameters and unnecessary computation during training/inference.

Denote the cellular signature is recorded as $M = (M^P, M^Q) \subset \mathbb{R}^{n \times 2}$, i.e., the RSRP and RSRQ readings heard from n cell towers (sorted by the signal strength), and our algorithm consists of three steps.

Step 1: Cellular signature normalization. We transform the measured cellular signature to a normalized value based on our bucket-based storing mechanism, and produce the corresponding signature vector V^P and V^Q. We also smooth such vectors with $\alpha = 0.1$ on its peak value.

Step 2: Similarity computation. For the ith heard cell tower, its RSRP and RSRQ signature vectors (Byte $1 \sim 7$) on grid j are represented as $F_{i,j}^P \subset \mathbb{R}^7$ and $F_{i,j}^Q \subset \mathbb{R}^7$ in the fingerprint set. We define the signature similarity as the dot product between two vectors, i.e.,

$$s_{i,j}^P = \frac{V^P \cdot F_{i,j}^P}{||F_{i,j}^P||_1}, s_{i,j}^Q = \frac{V^Q \cdot F_{i,j}^Q}{||F_{i,j}^Q||_1} \tag{4}$$

where $s_{i,j}^P$ and $s_{i,j}^Q$ represent the similarity on RSRP and RSRQ signatures, respectively.

Step 3: Central grid selection. In order to derive the most probable grid cell that the user may appear, we sum up the weighted similarity of each grid cell j over all heard cell towers, i.e.,

$$s_j = \sum_{i=0}^{n} w_i \cdot (\lambda^P \cdot s_{i,j}^P + \lambda^Q \cdot s_{i,j}^Q) \tag{5}$$

where $\lambda^P = 0.5$ and $\lambda^Q = 0.5$ represent the weight for RSRP and RSRQ signatures, respectively. w_i denotes the weight for individual cell tower based on its signal strength, e.g., $w_i = 1$ for the primarily-connected cell tower and $w_i = 0.8$ for the strongest neighbouring cell tower.

Finally, we derive the grid cell $\hat{j}$ with the highest score, and regard it as the central grid. We further construct a square area composed of $N \times N$ grid cells around the center grid as a candidate region (N is set as 32 based on the the large-scale experiments).

5.2 Feature Map Generation

Based on the candidate region, we aim to estimate user's relative position towards region center via current measurement data. Specially, we construct a multi-dimensional feature map on both primarily-connected and neighbouring cell towers via the large-scale fingerprint set. For grid cell j in candidate region, its features on cell tower i include four categories:

- RSRP signature similarity $s_{i,j}^P$,
- RSRQ signature similarity $s_{i,j}^Q$,
- PV heat feature h_j^{PV},
- UV heat feature h_j^{UV},

where RSRP(RSRQ) similarities are calculated via Equation 4, and PV(UV) features are computed via Equation 2. All of them are incrementally updated based on crowdsourced trajectories with a sliding time window (e.g., 30 days in our system).

Next, we normalize the values on each feature domain among all possible grids, and construct a multi-dimensional feature map of the candidate region. If there is no features in a grid, the feature value is set as zero. Finally, we stack $W = 4n$ feature maps among n cell towers as the input of CNN model.
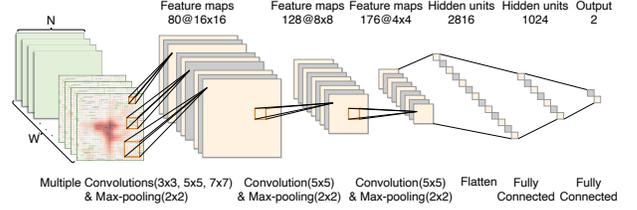
Figure 13: CNN structure, which consists of three convolutional layers, three max-pooling layers, and two fully connected layers. Feature maps are represented by *channel@height × width*.

5.3 CNN Model for Localization

Instead of directly estimating absolute coordinates, our CNN model predicts user's deviated location $(\Delta x, \Delta y)$ toward the center point (x_0, y_0) in candidate region with multi-dimensional feature maps.

Structure. As shown in Figure 13, our CNN model is comprised of three convolutional layers, three max-pooling layers, and two fully connected layers. In the first convolutional layer, we employ a multi-scale framework that utilizes three different convolution kernels (3×3, 5×5, and 7×7) to extract spatial features at varying scales. Then, a 5×5 convolution kernel is used for the last two convolutional layers. Also, three 2×2 max-pooling layers are leveraged to reduce network parameters. Finally, features are flatten and fed into two Fully Connected (FC) layers to predict the relative location.

Loss computation. Based on the predicted location deviation $(\Delta x, \Delta y)$, user's global coordinate $(\hat{x}, \hat{y})$ can be calculated by:

$$(\hat{x}, \hat{y}) = (x_0 + \Delta x, y_0 + \Delta y) \tag{6}$$

where (x_0, y_0) is the coordinate of the center point in candidate region. User's ground truth location (x, y) is measured by the opportunistic geo-tags, e.g., outdoor trajectories by the window. In addition, we employ the Haversine formula to compute the loss value, which implies the distance between the predicted location and its ground truth along earth surface, i.e.,

$$L = 2r \arcsin\left(\sqrt{\sin^2\left(\frac{\hat{y} - y}{2}\right) + \cos y \cos \hat{y} \sin^2\left(\frac{\hat{x} - x}{2}\right)}\right) \tag{7}$$

where r is the radius of the earth.

Implementation. Our models are implemented in TensorFlow. For training, we use the Adam optimizer with a learning rate of 5e-4 and a batch size of 512. These settings were determined through experimentation and optimization, ensuring the best performance for our cellular localization system. Training the model for 30,000 epochs allows it to learn and adapt to the specific characteristics of our dataset over China, enhancing its predictive capabilities.
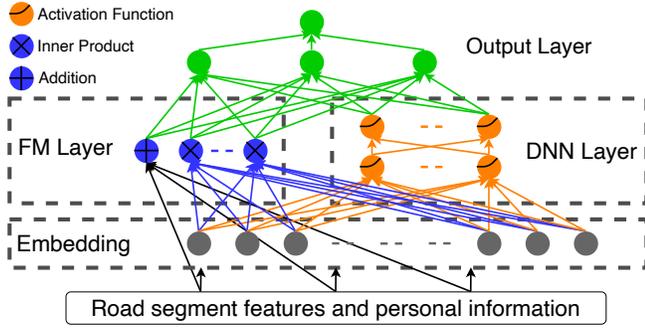
Figure 14: DeepFM architecture for pickup recommendation. It consists of an embedding layer, a FM layer, a DNN layer, and an output layer.

6 PICKUP RECOMMENDATION

With the estimated position by cellular localization at blackhole, DiDi ride-hailing platform provides an appropriate pickup position for each passenger. Specially, our pickup recommendation mechanism includes three steps, i.e., road discretization, candidate estimation, and customized recommendation.

Road discretization. Since passengers get on vehicles mostly by road sides, pickup positions should also be produced along the road side. In order to simplify the searching range, we divide roads into segments at $10m$ interval, and calculate the quantity of user visits on each road segment at different time.

Candidate estimation. First, we select all road segments within $500m$ to construct the candidate set, and extract the distance, user popularity, and historical trajectories on each candidate segment. In addition, we add individual passenger's pickup history within $1500m$ from the estimated position. Such personal information helps to ensure a minimal satisfactory service in case there sometimes exist extreme errors in cellular localization.

Customized recommendation. Traditional pickup recommendation at DiDi is formulated as a binary classification problem, i.e., whether the suggested pickup position is within tolerable region (less than $30m$). Our design has gone through two stages during its large-scale deployment.

Stage 1: Pairwise ranking instead of binary classification. Based on the crowdsourced large-scale orders, we propose a scoring mechanism to all candidate pickup locations to highlight the most confident ones, e.g., the suggested pickup position should be near and at the same road side with the practical one. Furthermore, we transform the binary classification issue into a pairwise ranking problem for better recommendation accuracy.

Stage 2: Deep neural model. Traditional recommendation in industry always adopt tree-based models, e.g. GBDT [14], but such model does not support online learning due to

Table 2: The relationship between pickup position error and call ratio/long call ratio.

Pickup Position Error	0~30m	30~50m	50~100m	>100m
Call Ratio	17.76%	33.28%	48.28%	69.87%
Long Call Ratio	0.74%	4.30%	10.61%	27.08%

its non-differentiable parameters, hence it can not adapt to dynamic orders which may extensively differentiate with the training set. In addition, tree-based models are also poor at capturing sufficient features with large amount of data. Thus, we explore and adopt a DeepFM [10] recommendation model.

The architecture of our DeepFM model is shown in Figure 14. Specially, it employs historical road features and personal information (e.g., user ID, city ID, week index, hour index) as inputs, and consists of four layers: 1) an embedding layer, transforms personal information into vectors, 2) a Factorization Machines (FM) [26] layer, learns linear and pairwise feature interactions, 3) a DNN layer, learns high-dimension features by deep neural networks, and 4) an output layer, combines FM layer and DNN layer to produce the final score.

7 LARGE-SCALE EVALUATION

7.1 Methodology

Data collection. Our large-scale cellular-based pickup service was deployed in practice with the DiDi application, a ride-hailing service like Uber. Our fingerprint dataset was collected and updated within 2 years, i.e., 2021-2022, across China. Our test dataset was also collected within these 2 years from the real deployed system, including ~50 million orders, ~13 million devices, and 918 brands of smartphones with ~9300 different models, across 4541 cities in China. We tested our system with all three main cellular service providers, i.e., China Mobile, China Unicom, and China Telecom.

It is worth to note that we only deployed our system on the Android platform as iOS provides its built-in cellular localization service. In other words, iOS users will use its built-in cellular localization service to get a position, and then use this position as the input for our pickup recommendation service. Finally, the pickup recommendation service gives a position and suggests users wait for the driver at that position.

Metrics. We compare our system from multiple aspects using the following metrics:

(1) **Pickup position error:** While academic research typically evaluates location accuracy using absolute location errors, it is impractical for the industry to cover the vast array of smartphones and the diversity of buildings in all cities. To enable large-scale evaluation, we adopt a crowdsensing approach that measures the
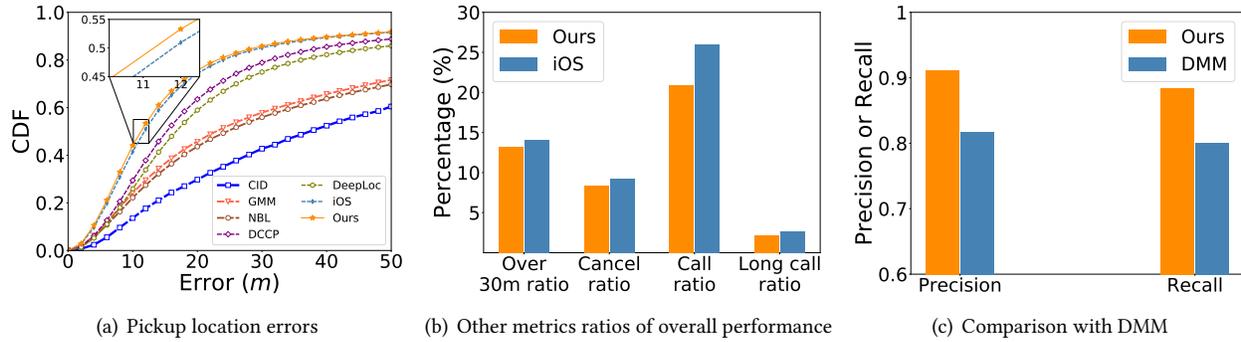
(a) Pickup location errors

(b) Other metrics ratios of overall performance

(c) Comparison with DMM

Figure 15: Performance across different scale of cities.

distance between recommended and actual pickup positions[4]. This distance metric is a key performance indicator (KPI) in the industry, as it helps avoid users making phone calls or extended conversations with drivers during the pickup phase (Table 2).

(2) **Over-30-meters ratio:** Our experience indicates that if the distance error is lower than 30 meters, the system will provide both drivers and passengers with satisfactory user experience because passengers typically don't want to move to another position on foot over-30-meters far from them. Therefore, we calculate the over-30-meters ratio as the ratio of the distance error higher than 30 meters with respect to the number of all the orders.

(3) **Cancel ratio:** If the system gives an incorrect position of the passenger to the driver, the driver cannot successfully find the passenger at that position. In this case, the passenger may cancel the order. Therefore, a lower cancel ratio means better system performance.

(4) **Call ratio:** Similarly, if the system gives an incorrect position of the passenger to the driver, the driver may make a phone call to ask where the passenger is. Therefore, a lower call rate indicates better system performance.

(5) **Long-call ratio:** Furthermore, drivers often keep calling until they successfully pick up the passenger. Therefore, if the driver stays calling longer than the $60s$, it means that the system gives a recommended position too far from the passenger. To this end, a lower long-call ratio means better system performance.

Comparisons. We consider the iOS built-in localization results as a system-level solution, as it can access physical information, while our approach is an application-level

solution using only public APIs[5]. We also compare our results with existing methods, including CID [34], GMM [8], NBL [22], DCCP [21], DeepLoc [31], and DMM [30].

7.2 Overall performance

Compared with iOS. The iOS system benefits from accessing physical information and unrestricted data at any time, providing a strong foundation for localization. On the other hand, our solution is an application-level approach relying on public APIs and data collected during app usage. Moreover, iOS devices have a homogeneous software and hardware environment, creating a controlled and stable ecosystem for localization. In contrast, Android devices come from numerous manufacturers, resulting in a wide variety of hardware sensors and software configurations. This diversity introduces significant instability in the collected fingerprints, making the localization task more challenging.

Our system outperforms iOS in terms of pickup location, as shown in Figure 15(a). Specifically, our system achieves a 0.54m (4.58%) lower median distance error compared to iOS. Additionally, our system exhibits lower over-30-meters ratio, cancel ratio, call ratio, and long-call ratio, as illustrated in Figure 15(b). This noteworthy accomplishment represents a modest improvement over the iOS native mechanism.

Compared with alternatives. Figure 15(a) demonstrates that our system achieves significantly better results compared to various alternatives. Specifically, the distance error of our system is 3.98m (26.15%) lower than DCCP, 13.34m (54.27%) lower than NBL, 11.69m (50.98%) lower than GMM, and 25.99m (69.81%) lower than CID. These improvements are attributed to our incorporation of user visits, cellular signatures, feature updating mechanism, and CNN model.

Regarding DeepLoc [31], which is designed for cellular localization in areas with fixed sizes and cell towers, we compare our approach with a specific area in a large city measuring $570m \times 510m$ with 114 cell towers obtained through

---

[4]The actual pickup position is recorded by drivers within the DiDi app when passengers board vehicles, as it marks the start of ride charging process.

[5]We are unable to implement our method on iOS devices because iOS does not provide developers with a publicly accessible API for obtaining signal strength information from servicing/neighboring base stations.
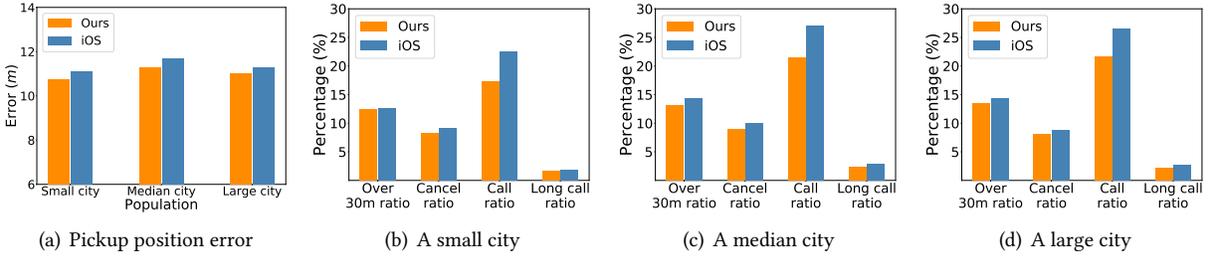
(a) Pickup position error (b) A small city (c) A median city (d) A large city

Figure 16: Performance across different scale of cities.

(a) Distribution (b) Distances (c) Other ratios
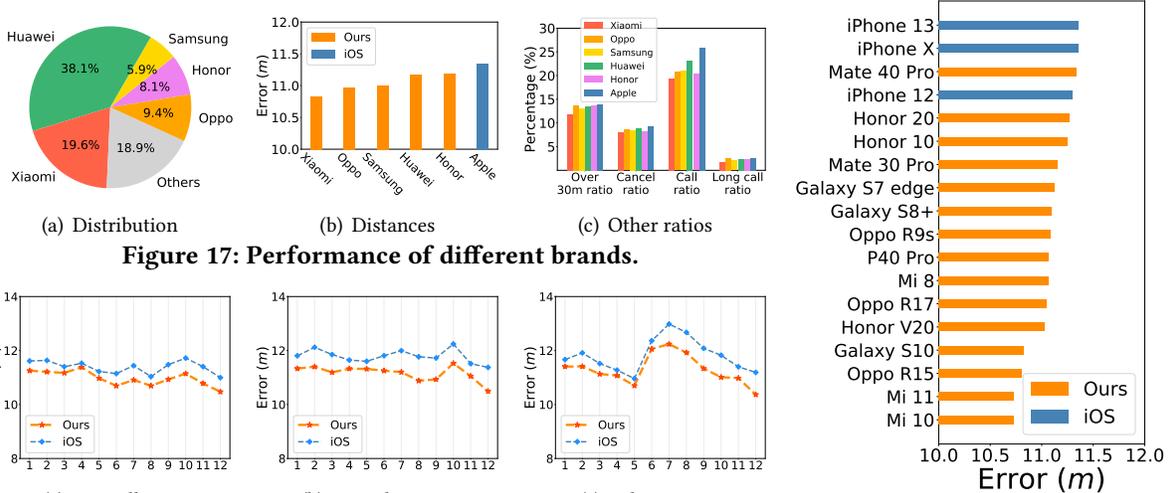
Figure 17: Performance of different brands.

(a) A small city (b) A median city (c) A large city
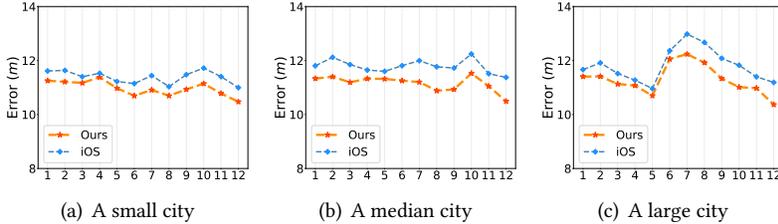
Figure 18: Performance among 12 months at three typical cities.

Figure 19: Performance comparison of different phone models.

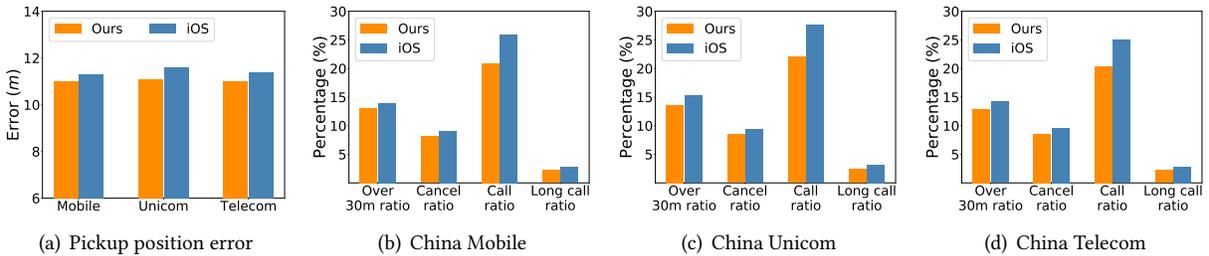(a) Pickup position error (b) China Mobile (c) China Unicom (d) China Telecom

Figure 20: Performance across different service providers.

crowdsensing. Our method achieves a significant reduction of 5.44m (32.63%) in median pickup position errors compared to DeepLoc which uses a multinomial logistic model.

Compared with DMM [30]. DMM is a map matching approach that operates on sequential cellular requests, while our method focuses on one-shot localization. Within DiDi, we have also developed and deployed a map matching method based on ListNet [6], Hidden Markov Model (HMM), and Deep Neural Networks (DNN). For comparison, we reproduce the DMM system and evaluate its performance using 200 real-world traveling sequences from our dataset, covering a distance of around 1,700 kilometers. Figure 15(c) shows that our approach has increased the precision by 9.43% (from 81.70% to 91.13%) and recall by 8.41% (from 79.99% to 88.40%),

thanks to the improved route connectivity and more accurate locations.

7.3 Impact of the city development level

Since different development levels of a city will have different environments, such as different heights of buildings, different numbers of base stations, etc., which might affect the fingerprint granularity and quality. Therefore, it is important to characterize our system performance in different cities with different development levels.

To characterize the performance of our system with different city development levels, we conduct experiments in three cities with different scales of population. In particular, we classify all the 4541 cities into three categories according to
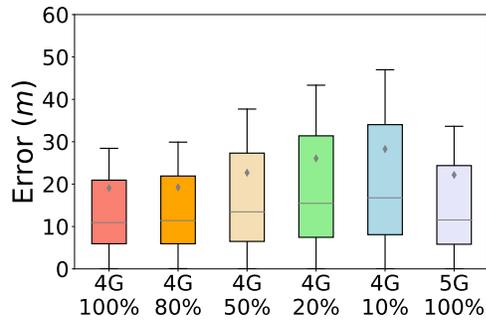
Figure 21: Comparison between 4G LTE network with different fingerprint density and 5G NR network.

their population, i.e., small (<10 million), median (10~20 million), and large (>20 million). Then we calculate our metrics in a typical city in each category, respectively.

Compared to iOS (Figure 16), our system gets lower pickup distance error, over-30-meters ratio, cancel ratio, call ratio, and long-call ratio, respectively. It indicates that our system will derive better user experience no matter the development level. In addition, the performance is found to be similar across the three cities, indicating the model's generalizability.

7.4 Impact of different brands and models of mobile phones

Since different brands or models of mobile phones will use different base-band chips, therefore, the cellular signal quality, strength, etc., and further the fingerprint might be different among different brands or models.

The distribution of different brands is shown in Figure 17(a). The top 5 brands of mobile phones in our dataset are Xiaomi, Oppo, Samsung, Huawei, and Honor. As shown in Figures 17(b) and 17(c), and Figure 19 we find that our system outperforms iOS in terms of all the metrics. On top of that, we compare the pickup position errors before and after deploying our system on four mainstream brands of Android smartphones: Mate 30 Pro, Mi 8, Oppo R17, and Honor V20. As shown in Figure 23, the deployed median errors have significant reduction of 15.33m (57.89%), 22.75m (67.29%), 16.34m (59.68%), and 37.77m (77.40%), respectively. These mean that our system performs well regardless of the brand and model of the smartphone.

7.5 Performance of fingerprint update along with the time

Since environmental changes will affect the fingerprint, the fingerprint should be updated within a certain period. To verify if our system performs stable along with the time and characterize the update mechanism of the fingerprint sampling, we calculate the metrics of 12 months separately

in the year 2022, with respect to different populations of cities.

As shown in Figure 18, our system deployed on the DiDi application outperforms the iOS-based system most of the time. Therefore, our system performs stably although the environment is changing. In terms of temporal variations, there is a difference of up to 2 meters in accuracy over the 12-month period, emphasizing the importance of incremental dataset updates.

7.6 Impact of different cellular service providers

Since different cellular service providers own their base stations at different positions, the fingerprints collected may be different among different providers. To verify if our system achieves a good performance regardless of different service providers, we calculate the metrics with different service providers, respectively.

As shown in Figure 20(a), our service deployed on the DiDi application achieves lower distance error compared to the iOS-based system. In addition, as shown in Figures 20(b), 20(c), and 20(d), our system could also achieve a better performance in terms of over-30-meters ratio, cancel ratio, call ratio, and long-call ratio. Therefore, our system could achieve a better performance regardless of cellular service providers.
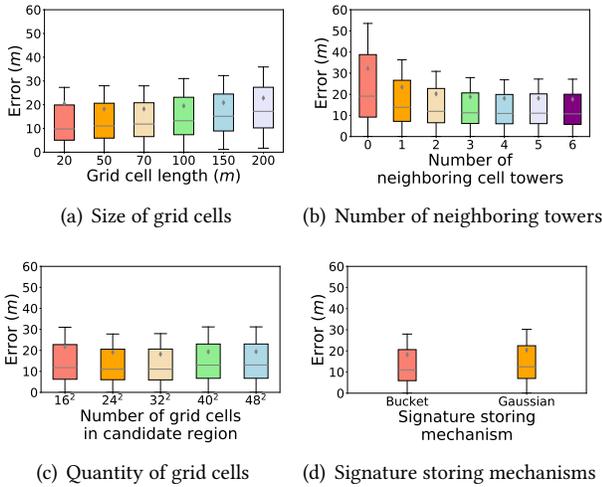
7.7 Comparison between 4G LTE and 5G NR

To evaluate the effect of 4G and 5G on cellular localization, we randomly select 100,000 travel orders within a week. Figure 21 illustrates that the median pickup position errors in 5G increase by 2.05m. This difference can be primarily attributed to the sparser availability of 5G fingerprints in our dataset. Specifically, we observe that the proportion of 5G mobile users is significantly lower (10.3%) compared to 4G users (89.7%).

To gain further insights, we conduct an experiment where we randomly remove some 4G fingerprints while keeping a comparable quantity of 5G fingerprints. Surprisingly, we find that the pickup position error in 4G increased by 5.67m. This result suggests that dense fingerprints and small cell coverage play a crucial role in improving cellular localization.

7.8 Effect of localization hyper-parameters and CNN features

Size of grid cells. Small grid cell size indicates fine-grained fingerprints on the map, but as a risk of dropping out of the candidate region with fixed number of grids. We compare the distance error of the size from $20m$ to $200m$. As shown in Figure 22(a), a box-plot diagram[6], large size causes more

---

[6]Whiskers extend from the box by 0.5x inter-quartile range (IQR) [1]

(a) Size of grid cells   (b) Number of neighboring towers

(c) Quantity of grid cells   (d) Signature storing mechanisms

Figure 22: Performance across different localization hyper-parameters. The horizontal line and the grey dot in the box represent the median error and the mean error, respectively.
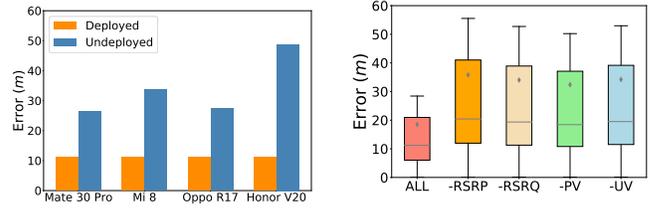

Figure 23: Deployed performance on four mainstream brands.

Figure 24: Ablation study on CNN features.

(a) Stage one: from binary classification to pairwise ranking   (b) Stage two: from GBDT to DeepFM

Figure 25: Performance of different pickup recommendation stages.

distance errors, while small size has better accuracy. However, we have noted that the mean distance error of $20m$ was worse than $50m$'s, even $70m$'s, which means that too small size could have larger uncertainty. Based on the above observations, we choose the grid cell length of $50m$ and continue the experiments.

Number of neighboring cell towers. Figure 22(b) depicts the effect of the number of neighboring towers used on the distance error. We clearly observe that using three or more neighboring cell towers could produce stable accuracy. In order to balance performance and robustness, we prefer to use 5 neighboring cell towers.

Quantity of grid cells. With such grid size, Figure 22(c) shows distance error with different number of grid cells to form the candidate region. With the increase of the number of grid cells, which means larger area of candidate region, the median and mean distance errors decrease first and then increase, and reach the optimal value when we choose $N = 32$ grid cells.
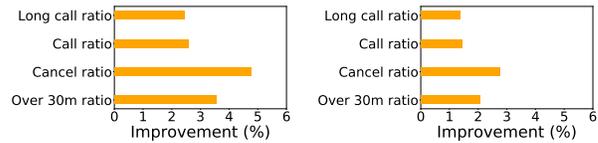
Signature storing mechanisms. Figure 22(d) shows the effect of our bucket-based storing mechanism on cellular signatures. Compared with Gaussian distribution, our design represents arbitrary distributions, thus it reduces median distance errors by $1.45m$, and mean distance errors by $2.23m$, respectively.

Ablation study on CNN features. To evaluate how much the addition of the four CNN features (i.e., RSRP, RSRQ, PV and UV) is actually helping, we perform an ablation study on each feature. Figure 24 demonstrates that using all four features provides the best accuracy.

7.9 Recommendation stage illustration

Recommendation stage one: from binary classification to pairwise ranking. When we prepare the dataset for training recommendation model, i.e., GBDT, there are two options. One is to conduct a binary classification task, which indicates that we only need to formulate positive and negative samples. The other option is to perform a pairwise ranking task, meaning that we should construct scoring samples. Figure 25(a) shows that the GBDT model trained by pairwise ranking obtains at least 2% improvements over the dataset trained by binary classification on all major service metrics, e.g., over-30-meters ratio, cancel ratio, call ratio and long-call ratio. Based on this result, we choose to construct and use the pairwise ranking dataset instead of binary classification.

Recommendation stage two: from GBDT to DeepFM. After adopting the pairwise ranking dataset, we contrast the impact of GBDT and DeepFM for recommendation on over-30-meters ratio, cancel ratio, call ratio and long-call ratio. Figure 25(b) indicates that the average percentage improvements of the four metrics are nearly 2% when we adopt DeepFM instead of GBDT, which shows that DeepFM learns abundant feature representations in the pickup position recommendation issue. Based on the above, we prefer to apply DeepFM in deployment.

8 RELATED WORK

Cellular-based localization. Compared with GPS technique, cellular network could provide ubiquitous localization for both indoor and outdoor environments [9]. First, Cell-ID systems [34] used the position of the strongest power cell

tower received by the UE as the estimated location of UE, which only offers a low-accuracy localization. Then, Angle of Arrival (AOA) [18, 23, 40] and Time of Arrival (TOA) [32] based methods were proposed with specific hardware for outdoor localization such as Unmanned Aerial Vehicle (UAV) localization [7].

Besides, fingerprint-based methods were developed since cellular stations were densely deployed [5, 9, 11, 16, 35]. For example, Cellsense system [13] used the measured RSSI values to estimate the user's location with a Bayesian-based method. Margolies et al. [22] leveraged 4G LTE to create a wide-area radio map and then developed a network-based localization with the fingerprinting method. Ray et al. [25] used Hidden Markov Model (HMM) and particle filter methods to obtain continuous trajectories. Chakraborty et al. [8] developed a geo-tag method using Gaussian mixture model (GMM) to reduce the impact of noise. Tian et al. [33, 39] proposed a subspace identification method, which could fully use internal relations of RF fingerprints to improve localization accuracy. In fact, the high localization accuracy in the above methods was still not guaranteed by using fingerprinting-based methods, because the large-scale data measurements and high-dimensional features were not fully exploited.

Deep learning-based localization. Compared with traditional machine learning methods (e.g., GMM, and HMM), deep learning [19] could effectively extract data features, which could address high-dimensional dataset and complex classification or regression tasks. Therefore, deep learning methods could be leveraged for indoor and outdoor localization with wireless data. For example, Wang et al. proposed the deep autoencoder network with channel state information (CSI) data for indoor localization [36]. More importantly, deep convolutional neural networks (DCNNs) were also used for CSI image-based and tensor-based indoor localization [37, 38], respectively. Moreover, adversarial learning was also used to address the environment change and heterogeneous device problems for indoor localization with CSI and RSSI values [4, 20]. Also, neural network was adopted to detect the building, floor, and location tracking based on RSSI signals [27, 28]. Currently, the light-weight neural network was also exploited to improve large-scale indoor localization performance with Bluetooth Low Energy (BLE) RSSI and geomagnetic field data [12].

Besides, deep learning techniques were also used for outdoor localization with cellular data. For example, DeepLoc [31] system focused on spatial and data augmentation techniques to reduce the calibration overhead in the training stage and to address the noise in RSSI and GPS data. However, this method only works in the small-scale outdoor environments. Also, DCCP [21] proposed a CNN method for outdoor localization with 4G LTE data. In addition, transfer learning was proposed for cellular outdoor localization with 2G Global

System for Mobiles (GSM) and 4G LTE Measurement Report (MR) datasets [42]. Moreover, deep reinforcement learning was used for fast map matching [30] with cellular data. Different of the previous work that directly predicts the final location, our *TransparentLoc* system first uses 5G NR to predict the user' relate position with CNN and then calculates the global coordinate, which can effectively reduce the localization errors in large-scale area.

9   DISCUSSION

Network switching. While jointly using both 4G and 5G measurement parameters can provide more comprehensive information, it may not be practical in practice. Most smartphones do not enable connectivity across multiple networks simultaneously, limiting their ability to detect cell towers from other mobile networks. Additionally, manual switching from 5G to 4G can be inconvenient for users and disrupt the overall user experience. As a result, sticking with the 5G network, especially if the signal strength is adequate, is often preferred over constant network switching.

Industrial service metrics. In order to minimize the impact of occasional cases such as order cancellations or phone calls, we conducted our evaluation with at least one million orders or more. This approach ensures that there is little fluctuation in each service metric, providing a more reliable assessment.

Integration with other approaches. Our method focuses on one-shot localization without the need for an indoor map. Integrating with indoor floorplans and other inertial tracking methods (e.g., Zee [24]) could lead to improved location accuracy and a more comprehensive localization solution.

10   CONCLUSION

In this paper, we share our technical insights and development experience to provide large-scale cellular localization availability for pickup service at DiDi ride-hailing platform. We address many practical challenges we encountered during the 2-year real-world deployment at most cities in China with millions of orders everyday. We hope this work can boost future attentions and efforts on cellular-based localization techniques which enables crucial safeguard services at anytime and anywhere. In future work, we will focus on the presence of abnormal cell towers that can potentially corrupt the cellular fingerprint dataset.

ACKNOWLEDGEMENT

REFERENCES

[1] 2022. Box-plot diagram. https://matplotlib.org/stable/api/_as_gen/matplotlib.axes.Axes.boxplot.html.

[2] 2022. *Network Survey Android App.* https://github.com/christianrowlands/android-network-survey

[3] Corin R Anderson, Pedro Domingos, and Daniel S Weld. 2001. Personalizing web sites for mobile users. In *Proceedings of the 10th international conference on World Wide Web*. 565–575.

[4] Roshan Ayyalasomayajula, Aditya Arun, Chenfeng Wu, Sanatan Sharma, Abhishek Rajkumar Sethi, Deepak Vasisht, and Dinesh Bharadia. 2020. Deep learning based wireless localization for indoor navigation. In *Proceedings of ACM MobiCom*. 1–14.

[5] Paramvir Bahl and Venkata N Padmanabhan. 2000. RADAR: An in-building RF-based user location and tracking system. In *Proceedings IEEE INFOCOM*, Vol. 2. 775–784.

[6] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*. 129–136.

[7] Ayon Chakraborty, Eugene Chai, Karthikeyan Sundaresan, Amir Khojastepour, and Sampath Rangarajan. 2018. SkyRAN: a self-organizing LTE RAN in the sky. In *Proceedings of the 14th International Conference on emerging Networking EXperiments and Technologies*. 280–292.

[8] Ayon Chakraborty, Luis E Ortiz, and Samir R Das. 2015. Network-side positioning of cellular-band devices with minimal effort. In *Proceedings of IEEE INFOCOM*. IEEE, 2767–2775.

[9] José A del Peral-Rosado, Ronald Raulefs, José A López-Salcedo, and Gonzalo Seco-Granados. 2017. Survey of cellular mobile radio localization methods: From 1G to 5G. *IEEE Communications Surveys & Tutorials* 20, 2 (2017), 1124–1148.

[10] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 1725–1731.

[11] Suining He and S-H Gary Chan. 2015. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys & Tutorials* 18, 1 (2015), 466–490.

[12] Yuming Hu, Feng Qian, Zhimeng Yin, Zhenhua Li, Zhe Ji, Yeqiang Han, Qiang Xu, and Wei Jiang. 2022. Experience: Practical Indoor Localization for Malls. (2022).

[13] Mohamed Ibrahim and Moustafa Youssef. 2012. CellSense: An Accurate Energy-Efficient GSM Positioning System. *IEEE Transactions on Vehicular Technology* 61, 1 (2012), 286–296.

[14] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems* 30 (2017).

[15] Ryan Keating, Mikko Säily, Jari Hulkkonen, and Juha Karjalainen. 2019. Overview of positioning in 5G new radio. In *Proceedings of IEEE International Symposium on Wireless Communication Systems (ISWCS)*. 320–324.

[16] Paulo Valente Klaine, Muhammad Ali Imran, Oluwakayode Onireti, and Richard Demo Souza. 2017. A survey of machine learning techniques applied to self-organizing cellular networks. *IEEE Communications Surveys & Tutorials* 19, 4 (2017), 2392–2431.

[17] Naresh Kumar, Rahul Rawat, and SC Jain. 2016. Bucket based data deduplication technique for big data storage system. In *2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*. IEEE, 267–271.

[18] Swarun Kumar, Stephanie Gil, Dina Katabi, and Daniela Rus. 2014. Accurate indoor localization with zero start-up cost. In *Proceedings of ACM MobiCom*. 483–494.

[19] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521, 7553 (2015), 436–444.

[20] Danyang Li, Jingao Xu, Zheng Yang, Yumeng Lu, Qian Zhang, and Xinglin Zhang. 2021. Train once, locate anytime for anyone: Adversarial learning based wireless localization. In *Proceedings of IEEE INFOCOM*. IEEE, 1–10.

[21] Yu Lin, Yao Tong, Qinkun Zhong, Ruipeng Gao, Buyi Yin, Lei Liu, Li Ma, and Hua Chai. 2021. DCCP: Deep Convolutional Neural Networks for Cellular Network Positioning. In *Proceedings of IEEE GLOBECOM*. 1–6.

[22] Robert Margolies, Richard Becker, Simon Byers, Supratim Deb, Rittwik Jana, Simon Urbanek, and Chris Volinsky. 2017. Can you find me now? Evaluation of network-based localization in a 4G LTE network. In *Proceedings of IEEE INFOCOM*. 1–9.

[23] Dragos Niculescu and Badri Nath. 2003. Ad hoc positioning system (APS) using AOA. In *Proceedings of IEEE INFOCOM*, Vol. 3. 1734–1743.

[24] Anshul Rai, Krishna Kant Chintalapudi, Venkata N Padmanabhan, and Rijurekha Sen. 2012. Zee: Zero-effort crowdsourcing for indoor localization. In *Proceedings of the 18th annual international conference on Mobile computing and networking*. 293–304.

[25] Avik Ray, Supratim Deb, and Pantelis Monogioudis. 2016. Localization of LTE measurement records with missing information. In *Proceedings of IEEE INFOCOM*. 1–9.

[26] Steffen Rendle. 2010. Factorization machines. In *2010 IEEE International conference on data mining*. IEEE, 995–1000.

[27] Hamada Rizk, Moustafa Abbas, and Moustafa Youssef. 2020. Omnicells: Cross-device cellular-based indoor location tracking using deep neural networks. In *Proceedings of IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 1–10.

[28] Hamada Rizk, Marwan Torki, and Moustafa Youssef. 2018. CellinDeep: Robust and accurate cellular-based indoor localization via deep learning. *IEEE Sensors Journal* 19, 6 (2018), 2305–2312.

[29] Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. 2017. DBSCAN revisited, revisited: why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems (TODS)* 42, 3 (2017), 1–21.

[30] Zhihao Shen, Wan Du, Xi Zhao, and Jianhua Zou. 2020. DMM: fast map matching for cellular data. In *Proceedings ACM MobiCom*. 1–14.

[31] Ahmed Shokry, Marwan Torki, and Moustafa Youssef. 2018. DeepLoc: A Ubiquitous Accurate and Low-Overhead Outdoor Cellular Localization System. In *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 339–348.

[32] Mohammed T Simsim, Noor M Khan, Rodica Ramer, and Predrag B Rapajic. 2006. Time of arrival statistics in cellular environments. In *Proceedings of IEEE 63rd Vehicular Technology Conference*, Vol. 6. 2666–2670.

[33] Xiaohua Tian, Xinyu Wu, Hao Li, and Xinbing Wang. 2019. RF fingerprints prediction for cellular network positioning: A subspace identification approach. *IEEE Transactions on Mobile Computing* 19, 2 (2019), 450–465.

[34] Emiliano Trevisani and Andrea Vitaletti. 2004. Cell-ID location technique, limits and benefits: an experimental study. In *Sixth IEEE workshop on mobile computing systems and applications*. IEEE, 51–60.

[35] Quoc Duy Vo and Pradipta De. 2015. A survey of fingerprint-based outdoor localization. *IEEE Communications Surveys & Tutorials* 18, 1 (2015), 491–506.

[36] Xuyu Wang, Lingjun Gao, Shiwen Mao, and Santosh Pandey. 2016. CSI-based fingerprinting for indoor localization: A deep learning approach. *IEEE Transactions on Vehicular Technology* 66, 1 (2016), 763–776.

[37] Xuyu Wang, Xiangyu Wang, and Shiwen Mao. 2018. Deep convolutional neural networks for indoor localization with CSI images. *IEEE*

Transactions on Network Science and Engineering 7, 1 (2018), 316–327.

[38] Xiangyu Wang, Xuyu Wang, and Shiwen mao. 2021. Indoor Finger-printing With Bimodal CSI Tensors: A Deep Residual Sharing Learning Approach. *IEEE Internet of Things Journal* 8, 6 (2021), 4498–4513.

[39] Xinyu Wu, Xiaohua Tian, and Xinbing Wang. 2018. Large-scale Wire-less Fingerprints Prediction for Cellular Network Positioning. In *Proceedings of IEEE INFOCOM*. 1007–1015.

[40] Jie Xiong and Kyle Jamieson. 2013. ArrayTrack: A Fine-Grained Indoor Location System. In *Proceedings of USENIX NSDI*. 71–84.

[41] Dongzhu Xu, Anfu Zhou, Xinyu Zhang, Guixian Wang, Xi Liu, Congkai An, Yiming Shi, Liang Liu, and Huadong Ma. 2020. Understanding op-erational 5G: A first measurement study on its coverage, performance and energy consumption. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 479–494.

[42] Yige Zhang, Aaron Yi Ding, Jörg Ott, Mingxuan Yuan, Jia Zeng, Kun Zhang, and Weixiong Rao. 2020. Transfer Learning-Based Outdoor Position Recovery With Cellular Data. *IEEE Transactions on Mobile Computing* 20, 5 (2020), 2094–2110.