

Real-world Large-scale Cellular Localization for Pickup Position Recommendation at Black-hole

Ruipeng Gao, Shuli Zhu, Lingkun Li, Xuyu Wang, Yuqin Jiang, Naiqiang Tan, Hua Chai,
Peng Qi, Jiqiang Liu, and Dan Tao

Abstract—Indoor localization availability is still sporadic in industry, especially at the black-hole, i.e., there only exist cellular signals, no GPS or WiFi signals. Based on our 2-year observations at the DiDi ride-hailing platform in China, there are 68k orders everyday created at black-hole. In this paper, we present *TransparentLoc*, a large-scale cellular localization system for pickup position recommendation of the DiDi platform. Specifically, we design a CNN model for real-time localization based on a crowdsourcing fingerprint set constructed by outdoor trajectories and abnormal cell tower detection. Then we leverage a DeepFM model to recommend an optimal pickup position for passengers. We 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, temporal variations, and abnormal cell towers in terms of four major service metrics, i.e., pickup position error, over-30-meters ratio, cancel ratio, and call ratio. The large-scale evaluations show that our system achieves a 0.54m lower median pickup position error compared to the iOS built-in cellular localization system, regardless of environmental changes, smartphone brands/models, time, and cellular providers. Additionally, the over-30-meters ratio, cancel ratio, and call ratio have significant reductions of 0.88%, 0.88%, and 5.13%, respectively.

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

I. 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 [2]. However, many sites, such as underground parking lots at airports or subway stations, lack the necessary infrastructure, like WiFi routers or ultrasonic speakers, to provide indoor localization services.

In this paper, we share our two-year experience providing accurate pickup services to passengers on DiDi, a prominent

A portion of this work was published in ACM MobiCom'23 [1] proceedings. This work was supported in part by NSFC Grant (62072029, 62402028, 62472023, 62402027), Beijing Nova Program (20230484263), Beijing NSF Grant (L221003), Fundamental Research Funds for the Central Universities (2022XKRC013, 2023YJS094), and DiDi Research Collaboration Plan. (Corresponding author: Lingkun Li.)

Ruipeng Gao, Shuli Zhu, Lingkun Li, Peng Qi, Jiqiang Liu, and Dan Tao are with Beijing Jiaotong University, Beijing, China. E-mail: {rpgao, zhushuli, lkli, pengqi1, jqliu, dtao}@bjtu.edu.cn.

Xuyu Wang is with Florida International University, Miami, USA. E-mail: xuyuwang@fiu.edu.

Yuqin Jiang, Naiqiang Tan, and Hua Chai are with DiDi, Beijing, China. E-mail: {jiangyuqin, tannaiqiang, chaihua}@didiglobal.com.

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. In addition, there may lack IT equipment such as WiFi access points in these areas, e.g., underground metro stations or parking structures, thus users solely rely 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 [3]), suffers from significant location errors, often in the range of hundreds of meters, due to the wide coverage of cell towers. Margolies *et al.* [4] leverage cellular signatures from multiple cell towers to mitigate environmental interference. Notably, DeepLoc [5] 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 New Ratio (NR) network, poses challenges. Cell towers' limited coverage and penetrability compared to network connectivity [6], 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. In addition, there exist a small number of abnormal cell towers with multiple clusters or impractical coverage area, which may result in coarse-grained positions.

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.

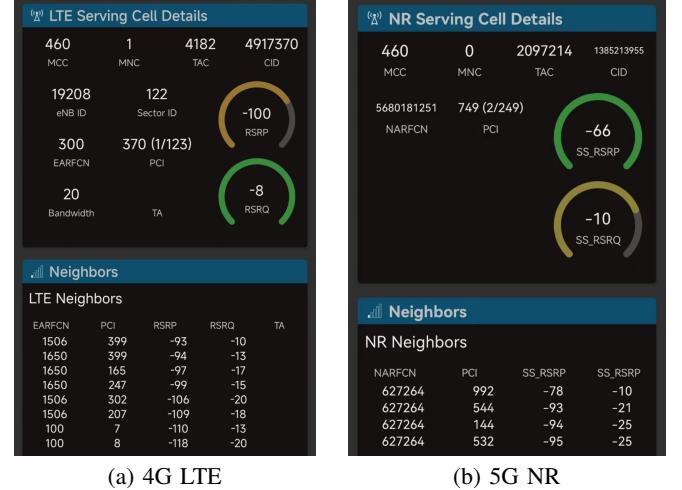
Third, we found that bits of abnormal cell towers produce huge confusion during localization phase, thus we utilized an unsupervised autoencoder model to identify such abnormal cell automatically.

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. Moreover, we have explored an unsupervised mining algorithm to eliminate negative effects of abnormal cell towers. 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



(a) 4G LTE

(b) 5G NR

Fig. 1: User Measurement Data (UMD) in 4G LTE and 5G NR network, recorded by a commodity app “Network Survey [8]”.

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.

The organization of the remainder is as follows. Section II introduces the challenges in cellular localization and our cell tower augmentation. Section III presents our system overview. Section IV and VI introduce fingerprint set construction and abnormal cell tower detection, respectively. Section V presents real-time localization. Section VII introduces pickup recommendation. Section VIII presents large-scale evaluation results of our system. Section IX introduces related work and Section X presents discussion. Finally, Section XI conclude this work.

II. CELL TOWER AUGMENTATION

A. Challenges in Cellular Localization

As shown in Figure 1, current smartphones can report User Measurement Data (UMD) in 4G Long-Term Evolution (LTE) networks, which includes wireless channel measurements like Reference Signal Received Power (RSRP), Received Signal Strength Indicator (RSSI), and Reference Signal Received Quality (RSRQ) [7]¹. 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.

¹In 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.

TABLE I: 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%

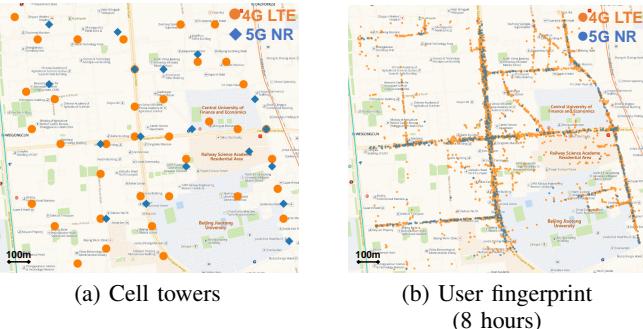


Fig. 2: Deployment of 4G LTE and 5G NR networks in the central region ($2\text{km} \times 2\text{km}$) in Beijing. The number of 4G cell towers are more than 2x of 5G (4G: 37, 5G: 18), and 4G user fingerprints are more than 5x of 5G (4G: 3550, 5G: 668).

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 [4]. 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 countries². This hinders the accuracy and robustness of cellular localization, limiting its usage mainly to emergency rescues. Below, we present observations from a $2\text{km} \times 2\text{km}$ 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 2a 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., $100 \sim 300\text{m}$) and weak penetrability, resulting in infrequent 5G signatures from neighboring towers. Based on our analysis with 1 million travel orders, Table I 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.

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

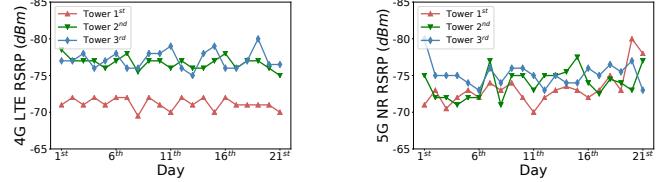


Fig. 3: Temporal variations on cellular signatures from the top 3 strongest cell towers over three weeks.

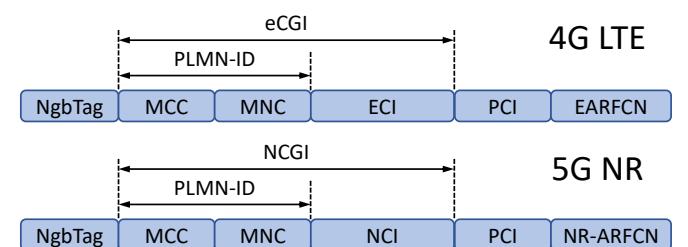


Fig. 4: Index construction for augmented cell towers.

(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 2b 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 shows that the RSRP measured in 5G was observed to be $-77 \pm 11.40\text{dBm}$, while it was measured as $-83 \pm 11.05\text{dBm}$ in 4G.

(4) Abnormal cell towers. From our long-term experience of the large-scale cellular localization service, we discover that there are a small number of abnormal cell towers that make positioning worse. For example, as we will see in Section V-A, extra-wide cell towers are sometimes able to cover the entire urban area, far beyond the coverage of normal cell towers, which may cause huge confusion in our localization phase.

B. Augmented Cell Tower as a Primitive

We aim to identify more available 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 E-UTRAN Cell Global Identifier (eCGI), which consists of the Public Land Mobile Network Identity (PLMN-ID) and E-UTRAN Cell Identifier (ECI), there are more stable and useful information such as PCI and EARFCN in 4G LTE network. Similarly, in 5G NR network, we can obtain PCI and NR-ARFCN besides the NR Cell Global Identifier (NCGI), which includes the PLMN-ID and the NR Cell Identifier (NCI).

TABLE II: The amount of “augmented” cell towers.

Augmented Cell Tower Unique Index	Amount
w/o PCI and EARFCN/NR-ARFCN	~ 0.04 Billion
w/o PCI	~ 0.30 Billion
w/o EARFCN/NR-ARFCN	~ 1.71 Billion
Ours	~ 2.20 Billion

Specifically, in LTE network PCI stands for Physical Cell ID, which is calculated by adding two different down link synchronisation signals, i.e., the primary synchronisation signal (PSS) and the secondary synchronisation signal (SSS) [9], [10]. The SSS (or PCI-group) consists of 168 sequence numbers: $N_{ID}^{(1)} \in [0, 167]$, and the PSS (or PCI-ID) consists of three different sequence numbers: $N_{ID}^{(2)} \in [0, 2]$. A PCI is defined as [11]:

$$N_{PCI} = 3N_{ID}^{(1)} + N_{ID}^{(2)}. \quad (1)$$

In addition, the EARFCN in 4G, which is similar to NR-ARFCN in 5G, denotes the Absolute Radio Frequency Channel Number. 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 and EARFCN (NR-ARFCN) together served as a part of the unique index for each “augmented” cell towers. Figure 4 provides the details of augmented cell tower unique index construction. NgbTag indicates whether the tower is a servicing one or a neighboring one. Table II provides the amount of augmented cell towers with different unique indexes. The results show that there exist approximately 2.20 billion “augmented” cell towers, leveraging both PCI and EARFCN/NR-ARFCN, which is $55\times$ the number of augmented cell towers without the use of PCI and EARFCN/NR-ARFCN. Specifically, 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 2a. This indicates the effectiveness to enrich the dimension of cellular signatures with more heard cell towers. In addition, Android does not ensure the ECI/NCI for neighbouring cell towers, thus our mechanism also increases the stability of discovering neighbouring cell towers in cellular localization.

III. SYSTEM OVERVIEW

Based on the augmented cell towers which provides sufficient network connectivity, we design the *TransparentLoc* system, which utilizes cellular measurements to ensure pick-up 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.

Aiming to effectively evaluate our system, we propose the performance metrics. Because covering numerous smartphones and buildings in all cities for industry is impractical, we design the pickup position error, namely, the distance between recommended and actual pickup positions. Additionally, we provide

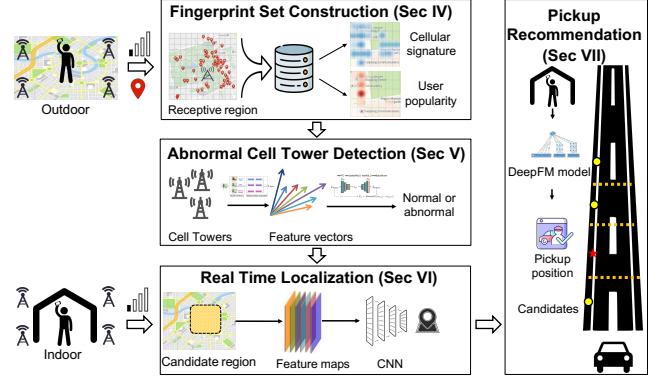


Fig. 5: *TransparentLoc* design at DiDi ride-hailing platform. It leverages outdoor trajectories with geo-tags to construct the cellular fingerprint set by crowdsourcing, automatically identify abnormal cell towers, and localizes indoor passengers by only cellular measurements. It further recommends a customized pickup position and informs both passengers and drivers.

other metrics, including the over-30-meters ratio, cancel ratio, call ratio, and long call ratio (see details in Section VIII).

Figure 5 shows the design overview of our system, which consists of four major phases: *fingerprint set construction* at outdoor with geo-tags, *localization with CNN* at indoor/outdoor by mere one-shot cellular measurements, *abnormal tower detection* by an unsupervised Encoder-based model, 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 with the size of $50m \times 50m$, and produces the receptive region³ 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 abnormal cell tower detection phase, we observe that there are a small number of abnormal cell towers which may provide huge errors for cellular localization, thus *TransparentLoc* identifies such abnormal cell towers based on a novel unsupervised detection algorithm.

In localization phase, *TransparentLoc* produces an appropriate candidate region by cellular measurements from the passenger’s mobile phone, which is a square area composed of 32×32 grid cells around the center grid cell. Then, we extract 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 candidate 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

³The “receptive region” refers to the geographical coverage area of a cell tower where it provides reliable cellular signal coverage to mobile devices.

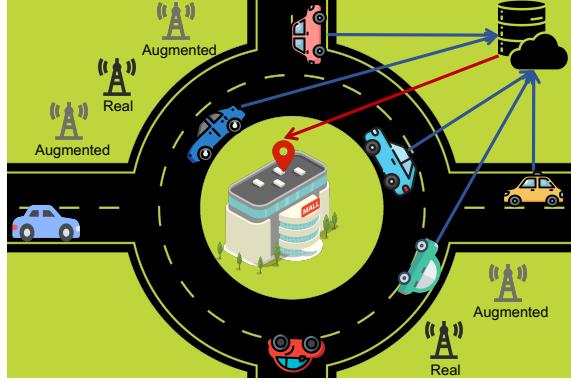


Fig. 6: Fingerprint set collection process.

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.

IV. FINGERPRINT SET CONSTRUCTION

The generalizability of the fingerprint set is crucial for industry-deployed learning-based cellular localization. It should cover 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 6 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. Note that, the errors of geo-tags are required to be less than 20m, which reflects the estimated horizontal accuracy radius in meters of this location at the 68th percentile confidence level from Android system API. Additionally, if the GPS locations with the errors that exceeds 20m, we will filter such inaccurate locations. Nevertheless, we still have enough accurate GPS locations to construct our fingerprint set.

Figure 7 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 7a). Samples within each grid cell are aggregated to produce specific 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 II-B) to calculate its receptive region at the grid level (Figure 7b). We further adopt the Density-Based

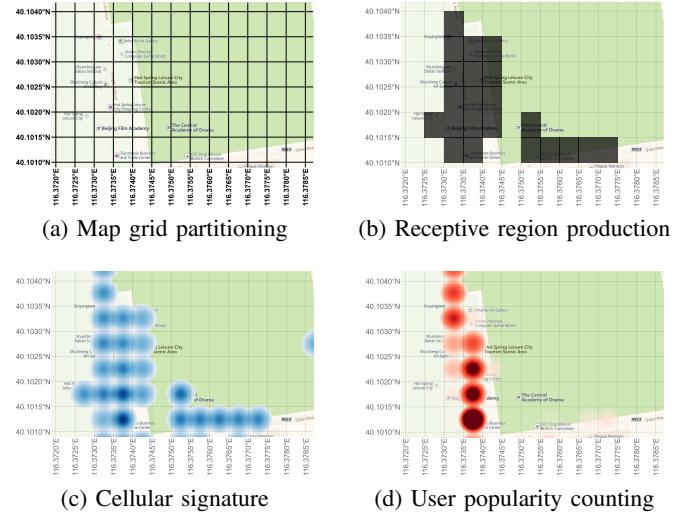
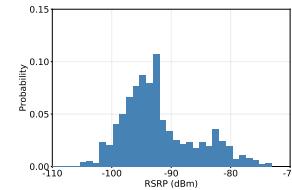
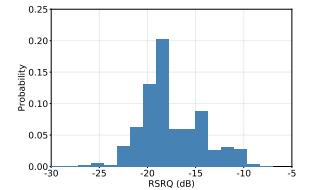


Fig. 7: Fingerprint set construction with four steps: map grid partitioning, receptive region production, cellular signature representation, and user popularity counting.



(a) RSRP distribution



(b) RSRQ distribution

Fig. 8: Arbitrary distribution of RSRP and RSRQ signatures heard from a 5G base station at a grid cell.

Spatial Clustering of Applications with Noise (DBSCAN) algorithm [12] 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 7c). Storing such vast data as-is incurs redundant storage and heavy computation. Existing approaches, like NBL [4], 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 8). To address this, we explore a bucket-based storing mechanism [13].

After profiling all historical RSRP and RSRQ signatures in China (Figure 9), 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 10). 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

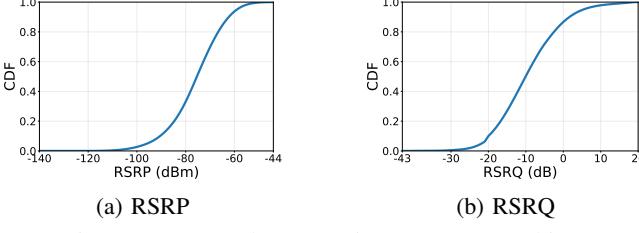


Fig. 9: RSRP and RSRQ signatures over China.

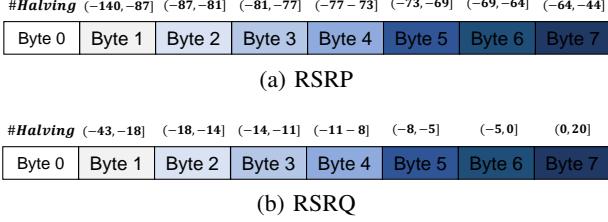


Fig. 10: Bucket-based storing mechanism on cellular signatures.

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 7d). Additionally, the user popularity varies dynamically in different date and time, especially during festivals or some big events. Figure 11 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 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 [14], i.e., Page View (PV) and Unique 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 $N_{j,t}^{UV}$ 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).

In sum, our fingerprint set is stored with a tree structure in database (Figure 12). The key for each “augmented” neighbouring cell tower is the unique index (elaborated in Section II-B), and followed by the grid Cell ID on the map. At each grid cell, its features (RSRP, RSRQ, PV and UV) are all incrementally updated everyday.

V. ABNORMAL CELL TOWER DETECTION

Based on our long-term experience of the large-scale cellular localization service, we observe that there are only bits

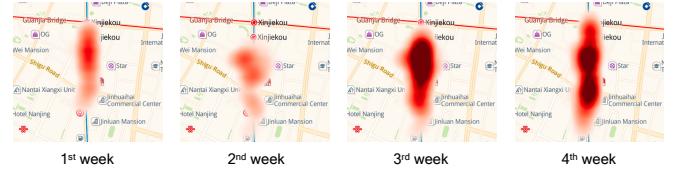


Fig. 11: User popularity in a shopping mall along four weeks in December.

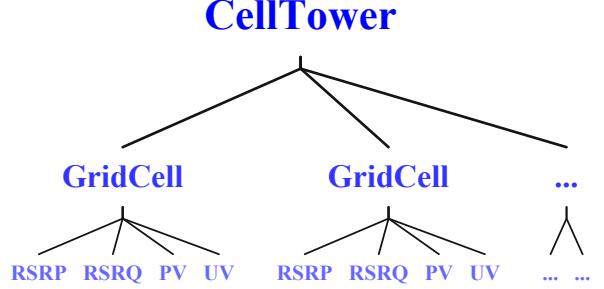


Fig. 12: Data format in fingerprint database.

of abnormal cell towers that may cause huge uncertainty in real-time localization phase. These abnormal cell towers are defined by their crowdsourced fingerprints. Thus, we investigate this issue and propose a detection algorithm to identify such abnormal cell towers.

A. Abnormal Cell Towers

As shown in Figure 13, we present some examples of abnormal cell towers during the deployment of cellular localization in DiDi ride-hailing platform. (1) **Multi-cluster cell towers:** Namely, cell towers have two or more clusters. An example in Figure 13a shows that there are three clusters in Chongqing of this cell tower. The system may locate passengers with huge errors due to the multiple clusters. (2) **Extra-wide cell towers:** Namely, the coverage area of extra-wide cell towers extremely exceed their actual coverage area. As illustrated in Figure 13b, the coverage area of this tower is over $220km^2$, while the coverage radius of cell towers in LTE network is less than $10km$. Consequently, localization with such extra-wide cell towers may result in a coarse-grained position, with a positioning error of more than $10km$ which is impractical.

B. Design Overview

Simple clustering algorithms, e.g., DBSCAN [12], which can identify multi-cluster cell towers. According to our statistics, the proportion of travel orders at black-hole with multi-cluster cell towers discovered by DBSCAN⁴ is around 1.15%. However, these clustering algorithms cannot accurately distinguish extra-wide cell towers. To overcome the limitations of clustering algorithms, we propose an unsupervised autoencoder model to discover these abnormal cell towers.

The key idea comes from the observation that there are much more normal cell towers than abnormal ones. The overall framework consists of two modules, i.e., *feature extraction* and

⁴The maximum distance between two samples for one to be considered as in the neighborhood of the other is set to 2,000 meters, while the number of samples in a neighborhood for a point to be considered as a core point is set to 3.

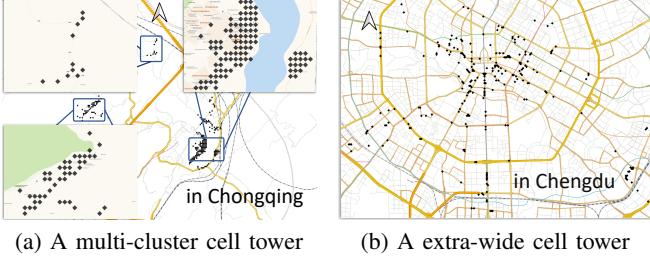


Fig. 13: Examples of abnormal cell towers. A black diamond represents a grid cell of this tower’s receptive region. (a) illustrates a cell tower has two far-away clusters, and (b) shows an extra-wide cell tower that covers over 220km².

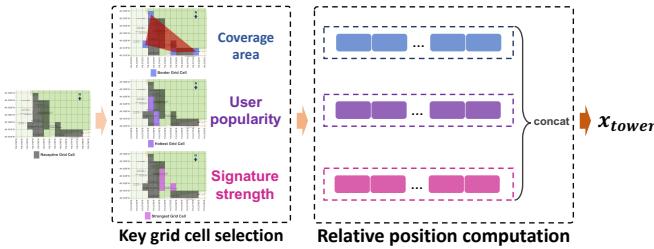


Fig. 14: Process of feature extraction. Firstly, we choose multiple key grid cells for each cell tower to represent their coverage area, user popularity, and signature strength. Secondly, we compute relative position of each key grid cell and then combine them in a fixed order to form a vector \mathbf{x}_{tower} . *unsupervised autoencoder detection*, which are described as follows.

C. Feature Extraction

We design a process to extract the fingerprint of each cell tower as a vector $\mathbf{x}_{tower} \in \mathbb{R}^{56}$. The details of feature extraction are shown in Figure 14.

(1) Key grid cell selection. We pick the most representative grid cells of the receptive region for each cell tower from three aspects, including coverage area, user popularity, and signature strength. In terms of **coverage area**, we choose the easternmost (rightward), southernmost (lower), westernmost (leftward), and northernmost (upper) grid cells as four boundary grid cells, respectively. If there are multiple grid cells on the border, we select the middle one among them. Regarding **user popularity**, we pick the top N_{kgc}^h hottest grid cells as key popularity cells, which reflect the most visited places around the cell tower. As regards **signature strength**, we select the top N_{kgc}^s grid cells with strongest cellular signals as key signature cells, which represent the signature distribution from the cell tower. In our system, we choose $N_{kgc}^h = N_{kgc}^s = 8$.

(2) Relative position computation. We calculate the relative positions of all the selected grid cells based on the geometric center of each cell tower. In addition, we arrange them in a fixed order to form the feature vector \mathbf{x}_{tower} for each cell tower, which will be considered as the input of the autoencoder network below.

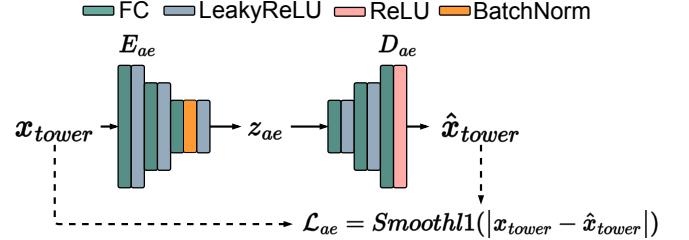


Fig. 15: Unsupervised autoencoder model structure. It consists of a encoder and a decoder, i.e., E_{ae} and D_{ae} . \mathbf{x}_{tower} stands for the feature vector of a cell tower extracted by *Feature Extraction* while $\hat{\mathbf{x}}_{tower}$ means a fake vector learned by the autoencoder model. \mathbf{z}_{ae} represents a latent vector.

D. Unsupervised Autoencoder Detection

We propose an unsupervised scoring mechanism to identify the abnormal cell towers, utilizing the autoencoder network with the feature vector \mathbf{x}_{tower} . Note that, “unsupervised” means that we do not need to prepare labeled samples of “abnormal” and “normal” cell towers for the autoencoder model training.

Structure. The structure of our unsupervised autoencoder detection model is illustrated in Figure 15, where E_{ae} and D_{ae} represent an encoder and a decoder, respectively. Based on the input, i.e., a cell tower vector \mathbf{x}_{tower} , the encoder E_{ae} learns a latent vector \mathbf{z}_{ae} that represents characteristic distribution of this cell tower, while the encoder D_{ae} generates the fake cell tower vector $\hat{\mathbf{x}}_{tower}$ using \mathbf{z}_{ae} .

Specifically, the encoder E_{ae} consists of three Fully Connected (FC) Layers with a LeakyReLU activate function and a Batch Normalization (BN) [15] layer. The number of hidden nodes in FC layers is 128, 64, and 32, respectively. The decoder D_{ae} is a combination of three FC layers with LeakyReLU and ReLU activate functions, and the count of hidden nodes in each FC layer is 64, 128, and 56, respectively.

Objective. The goal of our autoencoder model is to minimize the distance between \mathbf{x}_{tower} and $\hat{\mathbf{x}}_{tower}$. Let $\phi(\cdot; \mathcal{W})$ be an autoencoder network with set of weights \mathcal{W} . Given a training dataset $\mathcal{D}_n = \{\mathbf{x}_{tower}^1, \dots, \mathbf{x}_{tower}^n\}$, we define the autoencoder network objective as follows.

$$\begin{aligned} \min & \frac{1}{n} \sum_{i=1}^n \text{Smoothl1}(|\mathbf{x}_{tower}^i - \phi(\mathbf{x}_{tower}^i; \mathcal{W})|) \\ &= \frac{1}{n} \sum_{i=1}^n \text{Smoothl1}(|\mathbf{x}_{tower}^i - \hat{\mathbf{x}}_{tower}^i|), \end{aligned} \quad (4)$$

where $|\cdot|$ denotes the absolute value. And the loss function *Smoothl1* is defined as follows.

$$\text{Smoothl1}(\delta) = \begin{cases} 0.5\delta^2, & \text{if } \delta < 1 \\ \delta - 0.5, & \text{otherwise} \end{cases} \quad (5)$$

Considering there are abnormal cell towers, the L1 loss function would be more robust than L2 loss function.

Abnormal scoring. For a given vector $\mathbf{x}_{tower} \in \mathbb{R}^{56}$, we can naturally define the abnormal score s_{ae} , i.e.,

$$s_{ae}(\mathbf{x}_{tower}) = \text{Smoothl1}(|\mathbf{x}_{tower} - \phi(\mathbf{x}_{tower}; \mathcal{W}^*)|), \quad (6)$$

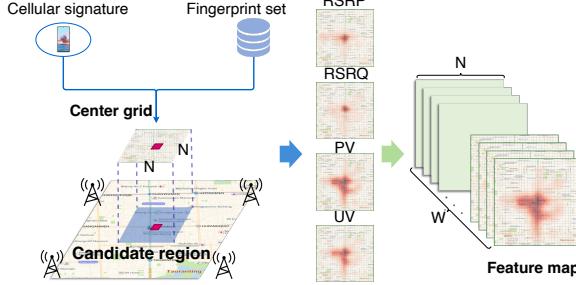


Fig. 16: Candidate region production and feature map generation.

where \mathcal{W}^* are the network parameters of a trained autoencoder model. A higher score indicates a higher probability of an abnormal cell tower. In our system, we identify an abnormal cell tower when its abnormal score exceeds our specified threshold. Specifically, we use the 97th percentile of abnormal scores of the last training set as “abnormal threshold”.

Implementation. The implementation of our model is based on the TensorFlow framework and we pick Adam with a learning rate of $1e-3$ and a batch size of 4096 as the optimizer. We randomly select approximately 8 million cell towers over China as the training dataset and the training epoch is set to 100.

VI. 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 16, our system consists of candidate region production, feature map generation, and CNN model training/inference.

A. 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 VIII-J). 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 i^{th} heard cell tower, its RSRP and RSRQ signature vectors (Byte 1 ~ 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.,

$$s_{i,j}^P = \frac{V^P \cdot F_{i,j}^P}{\|F_{i,j}^P\|_1}, s_{i,j}^Q = \frac{V^Q \cdot F_{i,j}^Q}{\|F_{i,j}^Q\|_1} \quad (7)$$

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.,

$$s_j = \sum_{i=0}^n w_i \cdot (\lambda^P \cdot s_{i,j}^P + \lambda^Q \cdot s_{i,j}^Q) \quad (8)$$

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).

B. 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 7, 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.

C. 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 17, our CNN model is comprised of three convolutional layers, three max-pooling layers, and two fully connected layers. In the first convolutional layer,

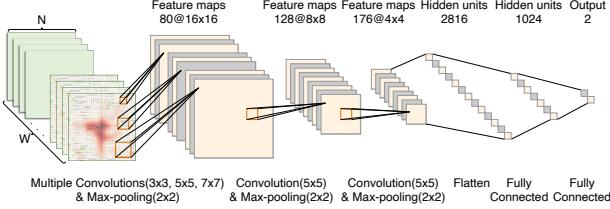


Fig. 17: CNN structure, which consists of three convolutional layers, three max-pooling layers, and two fully connected layers. Feature maps are represented by $channel@height \times width$.

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:

$$(\hat{x}, \hat{y}) = (x_0 + \Delta x, y_0 + \Delta y) \quad (9)$$

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.,

$$L = 2r \arcsin \left(\sqrt{\sin^2 \left(\frac{\hat{y} - y}{2} \right) + \cos y \cos \hat{y} \sin^2 \left(\frac{\hat{x} - x}{2} \right)} \right) \quad (10)$$

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. We train the model weekly and prepare approximately 10 million samples for each training. 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.

VII. PICKUP RECOMMENDATION

With the estimated position by cellular localization at black-hole, 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

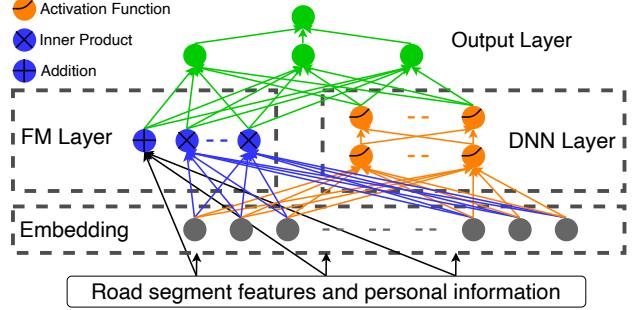


Fig. 18: DeepFM architecture for pickup recommendation. It consists of an embedding layer, a FM layer, a DNN layer, and an output layer.

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 [16], but such model does not support online learning due to 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 [17] recommendation model.

The architecture of our DeepFM model is shown in Figure 18. 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) [18] layer, learns linear and pairwise feature interactions, 3) a Deep Neural Network (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.

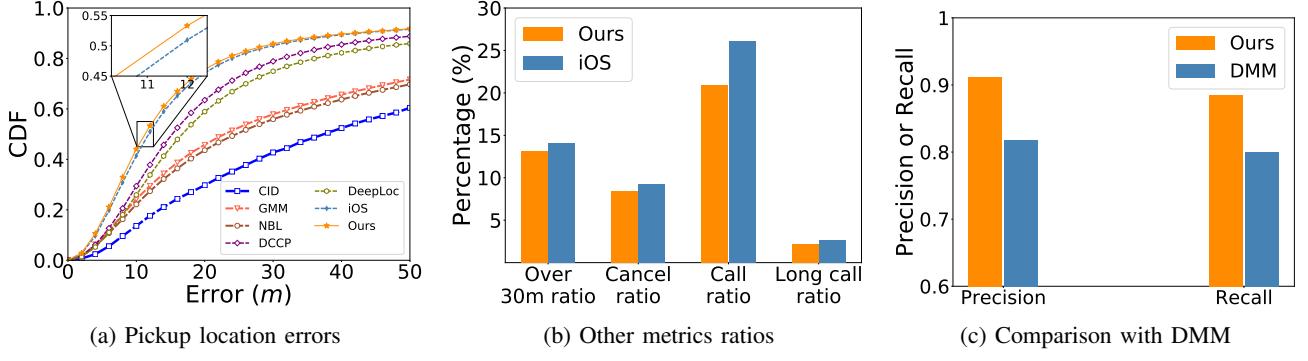


Fig. 19: Overall performance.

TABLE III: The relationship between pickup position error and call ratio/long call ratio.

Pickup Position Error	<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%

VIII. LARGE-SCALE EVALUATION

A. 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 distance between recommended and actual pickup positions⁵
- 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

⁵The actual pickup position is recorded by drivers within the DiDi app when passengers board vehicles, as it marks the start of ride charging process. 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 III).

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 APIs⁶. We also compare our results with existing methods, including CID [3], GMM [19], NBL [4], DCCP [20], DeepLoc [5], and DMM [21].

B. 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 19a. Specifically, our system achieves

⁶We 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.

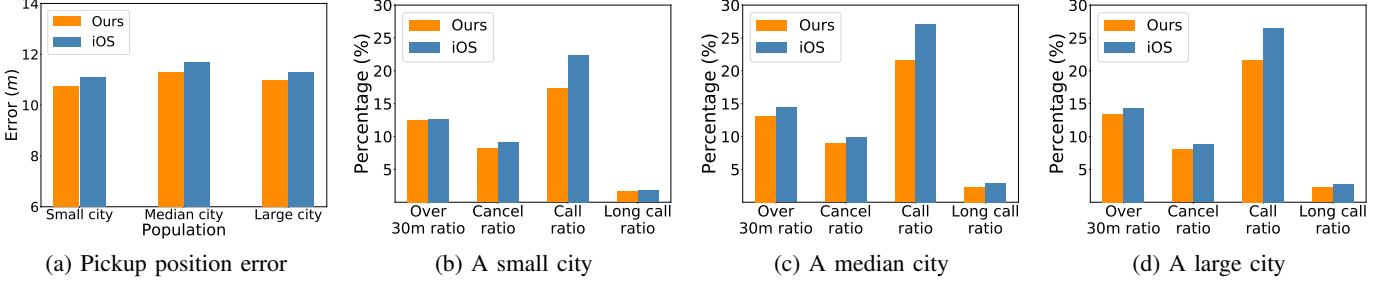


Fig. 20: Performance across different scale of cities.

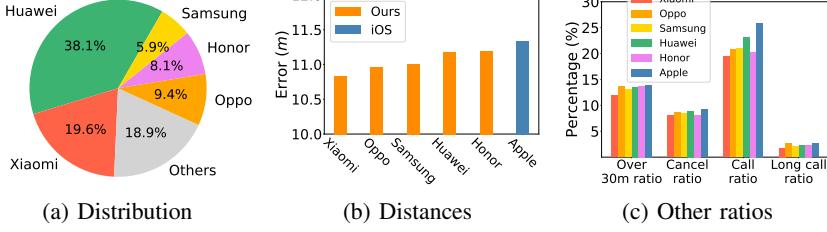


Fig. 21: Performance of different brands.

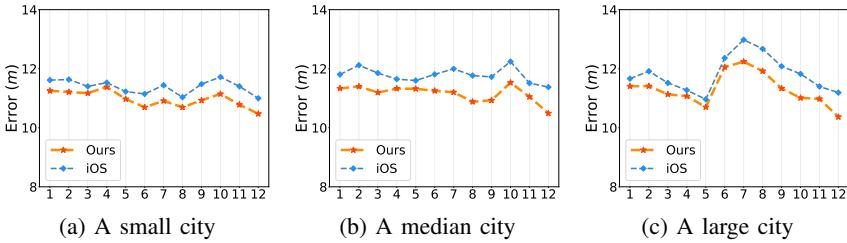


Fig. 22: Performance among 12 months at three typical cities.

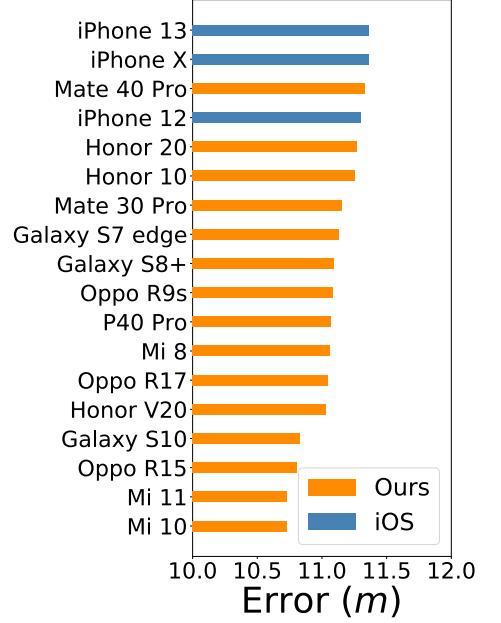


Fig. 23: Performance comparison of different phone models.

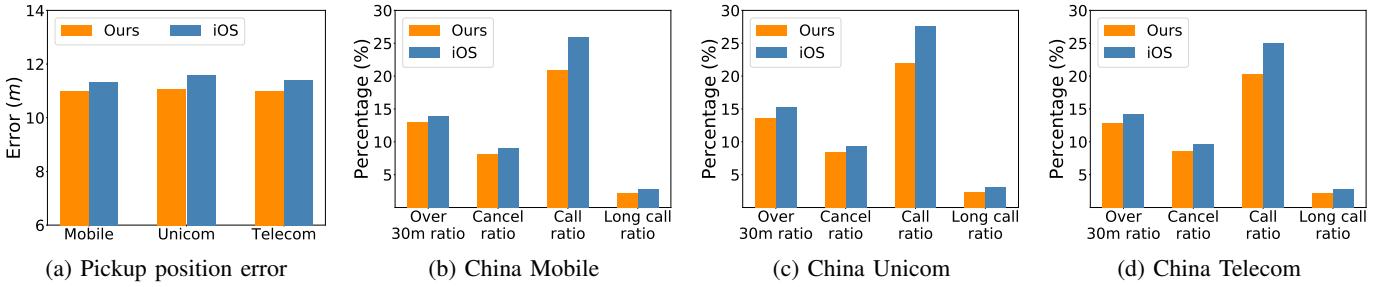


Fig. 24: Performance across different service providers.

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 19b. This noteworthy accomplishment represents a modest improvement over the iOS native mechanism.

Compared with alternatives. Figure 19a 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 signal

features, feature updating mechanism, and CNN model.

Regarding DeepLoc [5], 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 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 [21]. 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

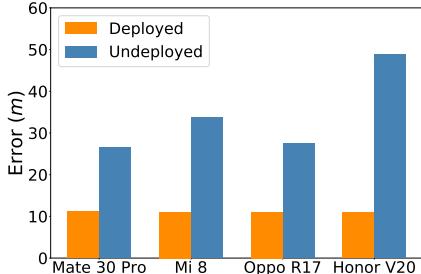


Fig. 25: Deployed performance on four mainstream brands.

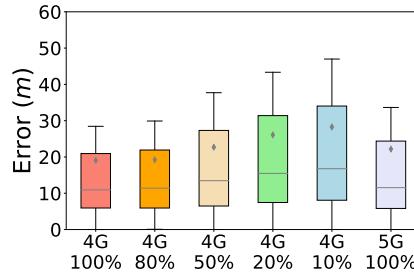


Fig. 26: Comparison between 4G LTE network with different fingerprint density and 5G NR network.

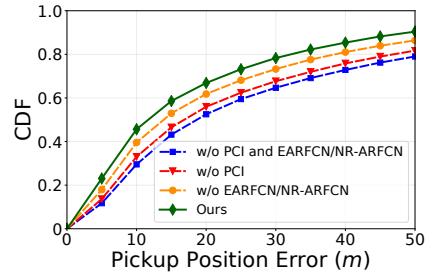


Fig. 27: Performance of using PCI and EARFCN/NR-ARFCN.

based on ListNet [22], 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 19c 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.

C. 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 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 20), 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.

D. 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 21a. The top 5 brands of mobile phones in our dataset are Xiaomi, Oppo, Samsung, Huawei, and Honor. As shown in Figures 21b and 21c, and Figure 23 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 25,

the deployed median errors have significant reductions of 15.33m (57.89%), 22.75m (67.29%), 16.34m (59.68%), and 37.77m (77.40%), respectively. These indicate the stability of our system, which have the significant ability to process the measurements from numerous Android smartphones regardless of the brands and models.

E. 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 22, 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, even the diversity of the collected fingerprints. 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.

F. 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 24a, our service deployed on the DiDi application achieves lower distance error compared to the iOS-based system. In addition, as shown in Figures 24b, 24c, and 24d, 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.

G. 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 26 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

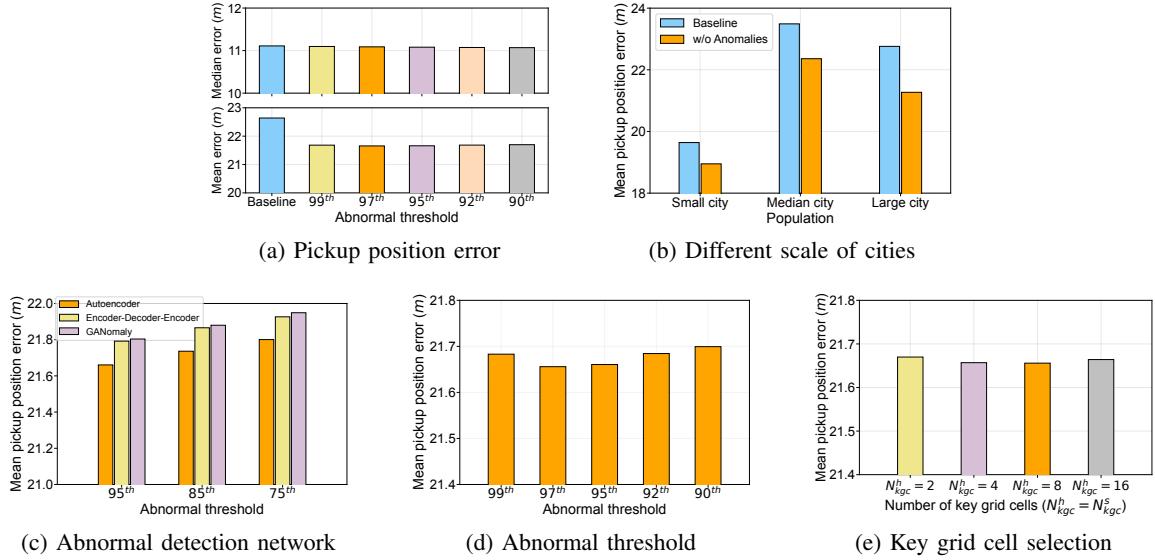


Fig. 28: Performance of abnormal cell tower detection.

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.

H. Impact of PCI and EARFCN/NR-ARFCN

To prove that using PCI and EARFCN/NR-ARFCN together in LTE/NR networks as a part of our augmented cell tower unique index plays a significant role in localization, we compare the performance of different augmented cell tower unique indexes in terms of pickup position error.

As shown in Figure 27, leveraging both PCI and EARFCN/NR-ARFCN achieves the best performance compared with using without PCI or EARFCN/NR-ARFCN. The results indicate that our augmented cell tower unique index construction is conducive to improve positioning accuracy.

I. Impact of abnormal cell tower detection

This sub-section aims to provide the evaluation of multiple aspects in abnormal cell tower detection phase (Section V).

Performance overview. In order to prove the effectiveness of our abnormal cell tower detection, we conduct experiments based on over one million travel orders which we randomly pick from daily orders. Specifically, we do not consider abnormal cell towers whose abnormal score exceeds the abnormal threshold in the real-time localization phase, except that all the scanned cell towers are identified as abnormal ones.

As shown in Figure 28a, we compare the median and mean pickup position errors with five different abnormal thresholds, i.e., the 99th, 97th, 95th, 92th, and 90th percentiles, respectively. The results in the upper subfigure illustrate that removing these abnormal cell towers discovered by the autoencoder model has no effect on the median pickup position

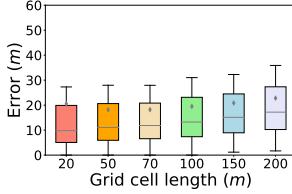
error. In contrast, the results in the bottom subfigure show that removing such abnormal cell towers reduces the mean pickup position error by 0.98m when the abnormal threshold is set to 97th percentile. These mean that filtering these detected abnormal cell towers can avoid huge location errors as much as possible.

Different scale of cities. To observe the performance of abnormal cell tower detection in different cities with different development levels, we randomly select three cities according to their population, i.e., a small city, a median city, and a large city. As depicted in Figure 28b, the results show that the abnormal cell tower detection achieves a better performance compared to the baseline. Specifically, the mean errors without abnormal cell towers have significant reductions of 0.69m, 1.13m, and 1.49m, respectively.

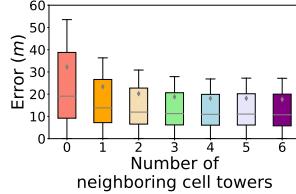
Abnormal detection network. There are several existing unsupervised networks for anomaly detection, such as autoencoder, GANomaly [23] (a GAN [24] based solution), and encoder-decoder-encoder network (the GANomaly without a discriminator). To derive the optimal network for abnormal cell tower detection, we evaluate their performance on the mean pickup position error. Figure 28c shows that the autoencoder network achieves the least pickup position errors with all abnormal identification thresholds.

Abnormal threshold. Aiming to search the best suitable abnormal threshold of the autoencoder model to detect abnormal cell towers, we compare the mean pickup position error utilizing different thresholds, namely, the 99th, 97th, 95th, 92th, and 90th percentiles. The results in Figure 28d show that the mean error decreases first and then increases, and reach the optimal value when we choose the 97th percentile as the abnormal threshold.

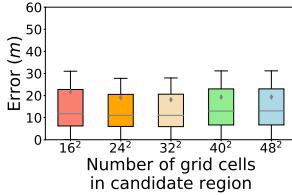
Key grid cell selection. A small number of key grid cells selected for anomaly detection may lead to the autoencoder model learning insufficiently. To find out the optimal number of key grid cells, we evaluate the autoencoder model with different N_{kgc}^h and N_{kgc}^s in terms of pickup position error. Figure 28e shows the mean pickup position error decreases



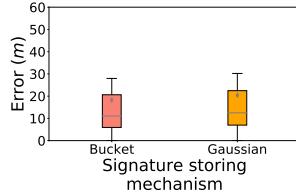
(a) Size of grid cells



(b) Number of neighboring towers



(c) Quantity of grid cells



(d) Signature storing mechanisms

Fig. 29: 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.

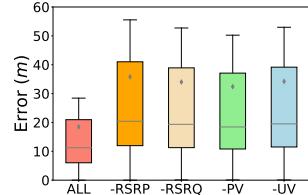


Fig. 30: Ablation study on CNN features.

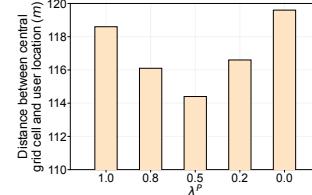


Fig. 31: Performance of the weights λ^P and λ^Q ($\lambda^Q = 1 - \lambda^P$)

first and then increase, and reach the optimal value when we choose $N_{kgc}^h = N_{kgc}^s = 8$.

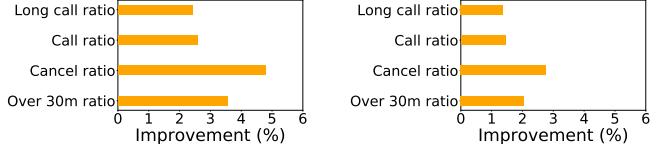
J. Effect of localization hyper-parameters and CNN features

This sub-section aims to provide the experiments about key hyper-parameters and CNN features selection in real time localization phase (Section VI).

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 29a, a box-plot diagram⁷, large size causes more 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 29b 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.

⁷Whiskers extend from the box by 0.5x inter-quartile range (IQR) [25]



(a) Stage one: from binary classification to pairwise ranking

(b) Stage two: from GBDT to DeepFM

Fig. 32: Performance of different pickup recommendation stages.

Quantity of grid cells. With such grid size, Figure 29c 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 29d 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 30 demonstrates that using all four features provides the best accuracy. Meanwhile, the results without RSRP achieve the worst performance, which indicates RSRP is the indispensable feature of CNN.

Weights of central grid selection. To find out the optimal values of λ^P and λ^Q in Equation 8, which represent the weight for RSRP and RSRQ signatures, respectively, we conduct experiments to compare the distance between central grid cell and user location with different values of λ^P and λ^Q . The smaller the distance, the more appropriate the values of λ^P and λ^Q . The results in Figure 31 show that the distance reaches the minimum value when $\lambda^P = \lambda^Q = 0.5$.

K. Recommendation stage illustration

This sub-section aims to provide the performance of two recommendation stages in pickup recommendation phase (Section VII).

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 32a 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 32b 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.

IX. RELATED WORK

A. Cellular-based localization

Compared with GPS technique, cellular network could provide ubiquitous localization for both indoor and outdoor environments [26]. First, Cell-ID systems [3] 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) [27]–[29] and Time of Arrival (TOA) [30] based methods were proposed with specific hardware for outdoor localization such as Unmanned Aerial Vehicle (UAV) localization [31].

Besides, fingerprint-based methods were developed since cellular stations were densely deployed [2], [26], [32]–[34]. For example, Cellsense system [35] used the measured RSSI values to estimate the user’s location with a Bayesian-based method. Margolies et al. [4] leveraged 4G LTE to create a wide-area radio map and then developed a network-based localization with the fingerprinting method. Ray et al. [36] used Hidden Markov Model (HMM) and particle filter methods to obtain continuous trajectories. Chakraborty et al. [19] developed a geo-tag method using Gaussian mixture model (GMM) to reduce the impact of noise. Tian et al. [37], [38] 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.

B. Deep learning-based localization

Compared with traditional machine learning methods (e.g., GMM, and HMM), deep learning [39] 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 [40]. More importantly, deep convolutional neural networks (DCNNs) were also used for CSI image-based and tensor-based indoor localization [41], [42], respectively. Moreover, adversarial learning was also used to address the environment change and heterogeneous device problems for indoor localization with CSI and RSSI values [43], [44]. Also, neural network was adopted to detect the building, floor, and location tracking based on RSSI signals [45], [46]. 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 [47].

Besides, deep learning techniques were also used for outdoor localization with cellular data. For example, DeepLoc [5] 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 [20] 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 [48]. Moreover, deep reinforcement learning was used for fast map matching [21] with cellular data. Different of the previous work that directly predicts the final location, our *TransparentLoc* system first uses LTE/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.

Additionally, TransLoc [49] presented a heterogeneous knowledge transfer framework for fingerprint-based indoor localization. Similarly, TRAIL [50] alleviated the problem of distribution discrepancy between online and offline data and differences among hetero-measure samples. In comparison, we avoid applying transfer learning since its data collection is not efficient for large-scale deployment with numerous buildings, and we use the crowdsourced outdoor trajectories to infer indoor locations.

C. Anomaly detection

Anomaly detection is a topic of significant interest in computer vision, with recent advancements leveraging generative adversarial networks (GANs). One influential approach, described in [51], assumes that the GAN’s latent vector represents the underlying data distribution and applies adversarial training for anomaly detection. Another notable model, called GANomaly, was introduced in [52]. It is a semi-supervised model that employs a conditional GAN with encoder-decoder-encoder sub-networks in the generator. GANomaly aims to learn the manifold of a large training dataset composed of normal samples and then detects anomalies within a testing dataset containing both normal and abnormal images. Researchers have also explored combining GAN-based anomaly detection with fingerprint-based localization. For instance, Haojun et al. developed RAD-GAN [53], an anomaly detection model for indoor fingerprinting-based localization inspired by GANomaly. RAD-GAN solely relies on unlabeled fingerprints and achieves outstanding performance, surpassing methods like OC-SVM [54]. Building upon these previous efforts, we propose an Encoder-based model for abnormal cell tower detection, treating it as an unsupervised learning problem where the boundary between normal and abnormal towers is not clearly defined. Unlike GAN-based methods, our model does not incorporate a discrimination network.

X. 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. We have noticed that passengers may cancel the order or make a phone call for various reasons, such as no need to travel, far-away driver locations, etc. 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. Additionally, we realize that user experience surveys are a valuable way to connect metrics with performance. However, we have not conducted such a well-designed survey over the past two years, and it is currently difficult to provide this data. Consequently, we hope we can conduct and analyze user experience surveys to find out the connection between metrics and performance clearly in the future.

Integration with other approaches. For instance, Zee [55] utilizes inertial sensors and indoor floorplans to track user continuous movements, while our method focuses on one-shot localization. In addition, the lack of indoor floorplans is one major obstacle to its ubiquitous coverage, since service providers have to conduct effort-intensive and time-consuming business negotiations with building owners or operators to collect the floor plans, or wait for them to voluntarily upload such data. Neither is conducive to large scale coverage in short time. In comparison, our solution does not rely on indoor floor plans, and integration with the above indoor tracking systems will further improve the location accuracy.

Abnormal cell tower classification. From our observations on abnormal cell towers, they possess two significant characteristics, i.e., multi-cluster and extra-wide. Clustering algorithms are conducive to detecting multi-cluster cell towers, while they are useless against extra-wide cell towers. In fact, as we remove abnormal cell towers in localization phase, it is not essential to classify these such abnormal cell towers into two or three categories.

System designed hyper-parameters. For the most of design parameters in *TransparentLoc*, we have conducted sufficient experiments to find their optimal values for our system. However, for the bits of hyper-parameters that are not vital for our system, we have designed them according to the long-term experience of cellular localization in the DiDi ride-hailing platform.

Areas without RF signals. It is worth noting that there exists indoor locations without cellular or WiFi signals due to signal attenuation, high-rise buildings blocking, etc, where we cannot provide cellular localization even the ride-hailing service. We will investigate pedestrian inertial tracking methods to improve the user experience.

XI. 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.

REFERENCES

- [1] S. Zhu, L. Li, X. Wang, C. Liu, Z. H. Yuqin Jiang, H. Chai, J. Liu, D. Tao, and R. Gao, “Experience: Large-scale cellular localization for pickup position recommendation at black-hole,” in *Proceedings of ACM MobiCom*, 2023, pp. 1270–1284.
- [2] P. Bahl and V. N. Padmanabhan, “Radar: An in-building rf-based user location and tracking system,” in *Proceedings IEEE INFOCOM*, vol. 2, 2000, pp. 775–784.
- [3] E. Trevisani and A. Vitaletti, “Cell-ID location technique, limits and benefits: an experimental study,” in *Sixth IEEE workshop on mobile computing systems and applications*. IEEE, 2004, pp. 51–60.
- [4] R. Margolies, R. Becker, S. Byers, S. Deb, R. Jana, S. Urbanek, and C. Volinsky, “Can you find me now? evaluation of network-based localization in a 4G LTE network,” in *Proceedings of IEEE INFOCOM*, 2017, pp. 1–9.
- [5] A. Shokry, M. Torki, and M. Youssef, “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*, 2018, p. 339–348.
- [6] D. Xu, A. Zhou, X. Zhang, G. Wang, X. Liu, C. An, Y. Shi, L. Liu, and H. Ma, “Understanding operational 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*, 2020, pp. 479–494.
- [7] R. Keating, M. Säily, J. Hulkko, and J. Karjalainen, “Overview of positioning in 5g new radio,” in *Proceedings of IEEE International Symposium on Wireless Communication Systems (ISWCS)*, 2019, pp. 320–324.
- [8] “Network Survey Android App,” 2022. [Online]. Available: <https://github.com/christianrowlands/android-network-survey>
- [9] M. E. Gujral, “Robust pci planning for long term evolution technology,” *EE Times*, 2013.
- [10] S. Nyberg, “Physical cell id allocation in cellular networks,” 2016.
- [11] A. ElNashar, M. A. El-Saidny, and M. Sherif, *Design, deployment and performance of 4G-LTE networks: A practical approach*. John Wiley & Sons, 2014.
- [12] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in *kdd*, vol. 96, no. 34, 1996, pp. 226–231.
- [13] N. Kumar, R. Rawat, and S. Jain, “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, 2016, pp. 267–271.
- [14] C. R. Anderson, P. Domingos, and D. S. Weld, “Personalizing web sites for mobile users,” in *Proceedings of the 10th international conference on World Wide Web*, 2001, pp. 565–575.
- [15] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *International conference on machine learning*. pmlr, 2015, pp. 448–456.
- [16] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, “Lightgbm: A highly efficient gradient boosting decision tree,” *Advances in neural information processing systems*, vol. 30, 2017.
- [17] H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, “Deepfm: a factorization-machine based neural network for ctr prediction,” in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2017, pp. 1725–1731.
- [18] S. Rendle, “Factorization machines,” in *2010 IEEE International conference on data mining*. IEEE, 2010, pp. 995–1000.

- [19] A. Chakraborty, L. E. Ortiz, and S. R. Das, "Network-side positioning of cellular-band devices with minimal effort," in *Proceedings of IEEE INFOCOM*. IEEE, 2015, pp. 2767–2775.
- [20] Y. Lin, Y. Tong, Q. Zhong, R. Gao, B. Yin, L. Liu, L. Ma, and H. Chai, "Dccp: Deep convolutional neural networks for cellular network positioning," in *Proceedings of IEEE GLOBECOM*, 2021, pp. 1–6.
- [21] Z. Shen, W. Du, X. Zhao, and J. Zou, "Dmm: fast map matching for cellular data," in *Proceedings ACM MobiCom*, 2020, pp. 1–14.
- [22] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li, "Learning to rank: from pairwise approach to listwise approach," in *Proceedings of the 24th international conference on Machine learning*, 2007, pp. 129–136.
- [23] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "Ganomaly: Semi-supervised anomaly detection via adversarial training," in *Computer Vision-ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III* 14. Springer, 2019, pp. 622–637.
- [24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.
- [25] "Box-plot diagram," https://matplotlib.org/stable/api/_as_gen/matplotlib.axes.Axes.boxplot.html, 2022.
- [26] J. A. del Peral-Rosado, R. Raulefs, J. A. López-Salcedo, and G. Seco-Granados, "Survey of cellular mobile radio localization methods: From 1g to 5g," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 2, pp. 1124–1148, 2017.
- [27] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AOA," in *Proceedings of IEEE INFOCOM*, vol. 3, 2003, pp. 1734–1743.
- [28] J. Xiong and K. Jamieson, "Arraytrack: A fine-grained indoor location system," in *Proceedings of USENIX NSDI*, 2013, pp. 71–84.
- [29] S. Kumar, S. Gil, D. Katabi, and D. Rus, "Accurate indoor localization with zero start-up cost," in *Proceedings of ACM MobiCom*, 2014, pp. 483–494.
- [30] M. T. Simsim, N. M. Khan, R. Ramer, and P. B. Rapajic, "Time of arrival statistics in cellular environments," in *Proceedings of IEEE 63rd Vehicular Technology Conference*, vol. 6, 2006, pp. 2666–2670.
- [31] A. Chakraborty, E. Chai, K. Sundaresan, A. Khojastepour, and S. Rangarajan, "SkyRAN: a self-organizing LTE RAN in the sky," in *Proceedings of the 14th International Conference on emerging Networking EXperiments and Technologies*, 2018, pp. 280–292.
- [32] Q. D. Vo and P. De, "A survey of fingerprint-based outdoor localization," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 491–506, 2015.
- [33] P. V. Klaine, M. A. Imran, O. Onireti, and R. D. Souza, "A survey of machine learning techniques applied to self-organizing cellular networks," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2392–2431, 2017.
- [34] S. He and S.-H. G. Chan, "Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 466–490, 2015.
- [35] M. Ibrahim and M. Youssef, "Cellsense: An accurate energy-efficient gsm positioning system," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 1, pp. 286–296, 2012.
- [36] A. Ray, S. Deb, and P. Monogioudis, "Localization of LTE measurement records with missing information," in *Proceedings of IEEE INFOCOM*, 2016, pp. 1–9.
- [37] X. Wu, X. Tian, and X. Wang, "Large-scale wireless fingerprints prediction for cellular network positioning," in *Proceedings of IEEE INFOCOM*, 2018, pp. 1007–1015.
- [38] X. Tian, X. Wu, H. Li, and X. Wang, "Rf fingerprints prediction for cellular network positioning: A subspace identification approach," *IEEE Transactions on Mobile Computing*, vol. 19, no. 2, pp. 450–465, 2019.
- [39] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [40] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763–776, 2016.
- [41] X. Wang, X. Wang, and S. Mao, "Deep convolutional neural networks for indoor localization with CSI images," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 1, pp. 316–327, 2018.
- [42] X. Wang, X. Wang, and S. mao, "Indoor fingerprinting with bimodal csi tensors: A deep residual sharing learning approach," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4498–4513, 2021.
- [43] R. Ayyalasomayajula, A. Arun, C. Wu, S. Sharma, A. R. Sethi, D. Vasisth, and D. Bharadia, "Deep learning based wireless localization for indoor navigation," in *Proceedings of ACM MobiCom*, 2020, pp. 1–14.
- [44] D. Li, J. Xu, Z. Yang, Y. Lu, Q. Zhang, and X. Zhang, "Train once, locate anytime for anyone: Adversarial learning based wireless localization," in *Proceedings of IEEE INFOCOM*. IEEE, 2021, pp. 1–10.
- [45] H. Rizk, M. Torki, and M. Youssef, "Cellindep: Robust and accurate cellular-based indoor localization via deep learning," *IEEE Sensors Journal*, vol. 19, no. 6, pp. 2305–2312, 2018.
- [46] H. Rizk, M. Abbas, and M. Youssef, "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, 2020, pp. 1–10.
- [47] Y. Hu, F. Qian, Z. Yin, Z. Li, Z. Ji, Y. Han, Q. Xu, and W. Jiang, "Experience: Practical indoor localization for malls," 2022.
- [48] Y. Zhang, A. Y. Ding, J. Ott, M. Yuan, J. Zeng, K. Zhang, and W. Rao, "Transfer learning-based outdoor position recovery with cellular data," *IEEE Transactions on Mobile Computing*, vol. 20, no. 5, pp. 2094–2110, 2020.
- [49] L. Li, X. Guo, M. Zhao, H. Li, and N. Ansari, "Transloc: A heterogeneous knowledge transfer framework for fingerprint-based indoor localization," *IEEE Transactions on Wireless Communications*, vol. 20, no. 6, pp. 3628–3642, 2021.
- [50] Y. Yang, X. Guo, C. Chen, G. O. Boateng, H. Si, B. Qian, and L. Duan, "Trail: A three-step robust adversarial indoor localization framework," *IEEE Sensors Journal*, 2024.
- [51] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *International conference on information processing in medical imaging*. Springer, 2017, pp. 146–157.
- [52] P. Perera, R. Nallapati, and B. Xiang, "Ocgan: One-class novelty detection using gans with constrained latent representations," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2898–2906.
- [53] H. Ai, T. Hu, and T. Xu, "Rad-gan: Radio map anomaly detection for fingerprint indoor positioning with gan," in *2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2021, pp. 1–8.
- [54] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural computation*, vol. 13, no. 7, pp. 1443–1471, 2001.
- [55] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proceedings of the 18th annual international conference on Mobile computing and networking*, 2012, pp. 293–304.



Ruipeng Gao received the B.S. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2010, and the Ph.D. degree from Peking University, Beijing, in 2016. He was a Visiting Scholar with Purdue University, West Lafayette, IN, USA, in 2019. He is a Professor with the School of Cyberspace Science and Technology, Beijing Jiaotong University, Beijing. His research interests include mobile computing and applications, the Internet of Things, and intelligent transportation systems.



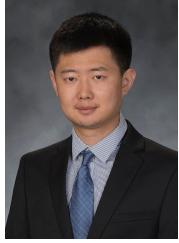
Shuli Zhu received the B.S. and M.S. degrees in software engineering from Beijing Jiaotong University, Beijing, China, in 2019 and 2021, respectively, where he is currently pursuing the Ph.D. degree in software engineering. His research interests include mobile computing and vehicle dead reckoning.



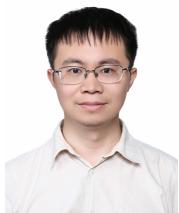
Lingkun Li is currently a lecturer at Beijing Jiaotong University. He received his Ph.D. degree in Computer Science from Michigan State University. His research interests include mobile computing, operating systems, and internet of things.



Peng Qi received the Ph.D. degree from the School of Computer Science, Beijing University of Posts and Telecommunications, Beijing, China, in 2022. He is currently with the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing. His research interests include data analysis, IoT and recommendation systems.



Xuyu Wang [S'13-M'18] received the M.S. in Signal and Information Processing in 2012 and B.S. in Electronic Information Engineering in 2009, both from Xidian University, China. He received the Ph.D. in Electrical and Computer Engineering from Auburn University, USA in 2018. He is currently an Assistant Professor in the Knight Foundation School of Computing and Information Sciences, Florida International University, USA. His research interests include wireless sensing, Internet of Things, wireless localization, smart health, wireless networks, and deep learning. He received the NSF CRII Award in 2021.



Yuqin Jiang received the B.S. degree in computer science from Tianjin University, China in 2011, and the M.S. degree in computer science from Zhejiang University, China in 2014. He is currently the Senior Algorithm expert with the Maps and Public Transportation Department, DiDi Company. His research interests include positioning, machine learning, and deep learning.



Jiqiang Liu received his B.S. (1994) and Ph.D. (1999) degree from Beijing Normal University. He is currently a professor at the School of Cyberspace Science and Technology, Beijing Jiaotong University. He has published over 200 scientific papers in various journals and international conferences. His main research interests are privacy-preserving, IoT Security, and cryptographic protocols.



Naiqiang Tan received the B.S. degree in Math from Hunan University, China in 2007, and the M.S. degree in computer science from Hunan University, China in 2010. He is currently the Distinguished Algorithm expert with the LLM Team, DiDi Company. His research interests include LLM, machine learning, and deep learning.



Dan Tao received the B.S. and M.S. degrees from Jilin University, China, in 2001 and 2004, and the Ph.D. degree from Beijing University of Posts and Telecommunications, China, in 2007. She was a Visiting Scholar in Illinois Institute of Technology, USA, from 2010 to 2011. She is currently a Professor with the School of Electronic and Information Engineering, Beijing Jiaotong University, China. Her research interests include IoT, crowdsensing, and intelligent information processing.



Hua Chai received the B.S. and M.S. degrees in computer science from Tianjin University, China, in 2006 and 2009, respectively. He is currently the Chair of DiDi Algorithm Committee, and the Head of DiDi IBG Technology & Customer Experience. He has rich experience in maps technology, internet advertising, large complex distributed systems, big data and machine learning.