Beauty or beast: human behavioral insights and learning power of federated mobility prediction

João Paulo Esper, Aline Carneiro Viana, Jussara M. Almeida

BEAUTY OR BEAST: HUMAN BEHAVIORAL INSIGHTS AND LEARNING POWER OF FEDERATED MOBILITY PREDICTION

A PREPRINT

João Paulo Esper
Department of Computer Science
Federal University of Minas Gerais, Belo Horizonte, Brazil
joaopauloesper@dcc.ufmg.br

Aline Carneiro Viana
Inria, Palaiseau, France
aline.viana@inria.fr

Jussara M. Almeida
Department of Computer Science
Federal University of Minas Gerais, Belo Horizonte, Brazil
jussara@dcc.ufmg.br

November 8, 2024

ABSTRACT

Mobility patterns are inherently linked to human nature (e.g., individual variability, temporal dynamics, behavioral factors, curiosity, social interaction), making mobility prediction a multifaceted and challenging problem that requires sophisticated models and comprehensive data. Machine learning (ML) models excel at predicting the location a person will be at the next time interval, but they often raise privacy concerns. To address these privacy issues while maintaining the benefits of ML models, Federated Learning (FL) offers a distributed framework that enables collaborative training of human mobility prediction models without requiring the sharing of highly sensitive location data. However, in the domain of FL for *individual* mobility prediction, prior work lacks a thorough understanding of the many factors that may impact the performance of FL-based prediction models. In this work, we provide a comprehensive study of the impact of various aspects related to human behavior, data characteristics, ML algorithmic solutions, and FL architectural structuring. We quantify the impact of such factors on effectiveness (accuracy) and efficiency (execution time, memory, and energy usage) of the prediction, revealing that, ignoring these factors lead to misleading result interpretation, and acknowledging them empowers both effectiveness and efficiency results.

1 Introduction

Mobility often reflects the inherent human *need to connect* with others (for family, work, or leisure) and is shaped by cultural norms and practices. Indeed, mobility is linked to *human nature* in various ways. For example, humans have a natural *curiosity* and desire to *explore*; they also *adapt to changing* environments and circumstances, leading to shifts in mobility patterns over time. Conversely, daily *routines and habitual behaviors* are a significant aspect of human nature, influencing the predictability of some mobility patterns, such as daily commutes. In such context, the prediction of mobility patterns correlates with a variety of applications aimed to improving human life, such as traffic forecasting [1], mobile network handovers [2], and epidemic control (e.g., COVID-19) [3]. Some applications rely on predicting mobility at a population level (e.g., traffic forecasting), while others focus on individual-level prediction. We here focus on the latter: given a history of locations visited by a person, the task is to predict the location she will be in the next time instant [4]. Previous work on individual mobility prediction has explored a wide range of techniques, from traditional Markov-based models [5, 6] to machine learning (ML) models, such as logistic regression [7] and, more recently, neural network models. The latter have often outperformed traditional approaches, especially in terms

of accuracy [8, 9, 10, 11]. Yet, even though accuracy is of great importance, there is a growing concern about how (private) data (e.g., visited locations) is collected and processed[1].

To meet those user concerns, new privacy-preserving machine learning approaches have been proposed to support the design of ML-based prediction models. One such approach is the Federated Learning (FL) framework [12], which has been applied to various domains, including Internet of Things (IoT) [13], healthcare [14] and Natural Language Processing (NLP) [15]. FL offers privacy preservation while still achieving competitive ML results. To achieve that, a shared model is trained in a *decentralized* fashion on multiple devices, independently, i.e., private user data is only stored and modeled locally on the user's device, creating *local models*. These local models (indeed, the model parameters) are then uploaded to a remote server that aggregates them to form a *global model*, typically using methods like averaging. The global model is then redistributed to devices. This process is repeated iteratively until the model converges. The key privacy advantage of FL lies in keeping private user data on local devices. Only the learned local models (i.e., model parameters) are shared with the remote server, significantly reducing the risk of data breaches [12].

Yet, applying FL to individual mobility prediction remains a significant challenge. The literature [16, 11, 17, 18, 19, 20] primarily focuses on improving mobility prediction accuracy. More recently, some studies have started addressing the heterogeneity of clients' mobility patterns in the design of local and global models [21, 22]. While these studies offer valuable insights, they mainly consider heterogeneity in terms of data sampling diversity or model parameter dissimilarity. Furthermore, like other prior studies, they focus on aggregated results, which can obscure performance differences that emerge for users with diverse mobility profiles. Overall, existing FL literature on individual mobility prediction lacks a *comprehensive understanding of the factors that most impact FL prediction results*, including aspects ranging from human behavior to solution design.

Our present goal is to fill this gap by offering a much broader and insightful evaluation of using the *FL framework for individual mobility prediction* (or FL4iMP for short). A key research question driving our study is: *"How do various factors associated with the FL4iMP environment, from the human nature of its participants to architectural and algorithmic design decisions, impact its performance?"* Specifically, we aim to address: *"How does FL4iMP perform for people with very diverse mobility patterns?"*.

To answer these questions, we identify different factors impacting FL4iMP performance (cf. Section 3). On the one hand, individual mobility is naturally heterogeneous, as it is driven by contextual aspects, personal characteristics, and motivations [23, 24, 25, 26, 27, 28, 29]. Thus, *human* factors related to individual mobility are key to our study. Additionally, for prediction purposes, human mobility is captured by the available data used to train (and test) the prediction models. As such, different characteristics of the *data* may impact the prediction results. Diving deeper into the FL4iMP solutions, factors associated with the type of learning *algorithm* used to build the local and global models, as well as how the federation *architecture* is established, may also impact the FL execution and results. To our knowledge, prior studies of FL4iMP solutions are quite restricted when evaluating such factors, either by limiting the scope of exploration or by neglecting some aspects, notably those related to individual behavior and data characteristics.

With that in mind, we aim at assessing how different factors associated with *human* behavior, *data* properties, learning *algorithms* as well as the FL *architecture* impact the FL4iMP results. Our study relies on previously proposed features [30, 25, 31, 28, 32] and a profiling approach [23, 33] to identify and characterize different mobility patterns at the individual level (cf. Section 4). Additionally, our analyses are performed on two real-world cellular datasets, the *Shanghai* and *Shenzhen* datasets [34], which have very different properties in terms of spatial and temporal coverage as well as population and area diversity. We rely on *Flower*, a state-of-the-art framework for FL [35] to instantiate multiple FL4iMP solutions, by using two obtained literature FL4iMP models [19] and varying architectural parameters, within a unified (and thus comparable) setup (cf. Section 5). Our evaluation considers metrics related to both effectiveness (prediction accuracy) and efficiency (execution time, memory, and energy usage). In sum, compared to prior work, we offer a much more comprehensive study of FL4iMP, providing insights into its performance in various relevant scenarios.

Our research shows that *Human* and *Data* factors have the most significant impact on the performance of FL4iMP models, with accuracy ranging from 0.297 to 0.867 within the same dataset, even with identical FL4iMP model and hyper-parameters. *Algorithm* and *Architecture* factors underscore the complexity of interpreting both efficiency and effectiveness results of FL4iMP models. Finally, we provide guidelines for FL4iMP architecture design, for algorithm choice, appropriate data selection and processing, and most importantly, data distribution across clients, aiming to avoid misleading results and enhance performance.

[1] e.g., GDPR in Europe (https://gdpr-info.eu/).

2 Setting the FL stage

In this section, we set the stage for the discussion of FL for individual mobility prediction (FL4iMP) by first presenting its general operation. Then, we discuss the existing literature on individual mobility prediction, with focus on models based on FL. For presentation purposes, throughout the rest of the paper, we use the terms individual and user interchangeably.

2.1 FL4iMP Ecosystem

A FL4iMP solution is fed by traces of data describing the mobility trajectories of users. Each trajectory $Traj^u = \{l_1^u, l_2^u, ..., l_T^u\}$ represents a time-ordered sequence of locations l_i^u user u visited while moving around a target geographical area. For decentralized learning of the prediction model, user trajectories are stored in local devices and given as the main input to the FL4iMP solution. The goal of the FL4iMP model is then to predict the next location l_{T+1}^u where the user u will be at the timestamp $T + 1$.

The workflow is as follows: **1.** Each device trains a *local model* using only the local (*training*) data for a certain *number of epochs*. The *number of epochs* determines how many times each device will iterate over its entire local dataset during this local training step. **2.** Instead of uploading their sensitive data, the devices only share the results of their locally trained models with the remote server, e.g., model parameters or weights. **3.** The server aggregates multiple model weights/parameters, building a *global model*. While different aggregation strategies have been proposed [22, 36], *FedAvg* [12] is the most commonly used one, which averages local model parameters to update the global model. **4.** The server distributes the *global model* to local devices. The previous steps are repeated for a certain number of *communication rounds*. The *communication rounds* refer to the exchange of model updates between the server and the clients. This iterative process continues until **5.** a *stop criterion* is met (e.g., convergence). The learned model can then be used to predict the next location visit of a (test) user u (l_{T+1}^u) [11].

Note that, as described, only the (local) model parameters are shared with the remote server, while the user's sensitive data (i.e., visited locations) are kept at the local devices. This fundamental property of FL makes it an attractive framework to address the increasing user concerns with privacy protection [12]. Indeed, in an *ideal private scenario*, each user runs her own local model on her own local device, protecting her data from others. However, often due to hardware limitations, prior studies on FL adopt a different architecture [20, 19, 37], using the abstraction of *data silos*, i.e., local entities that aggregate data from multiple users (e.g., users within the same department [38]). Throughout this paper, we use the term *client* to refer to such local entities, which could be either personal devices or data silos aggregating multiple users.

2.2 Literature review

To our knowledge, one of the first FL4iMP solutions is *PMF* [11]. One of its key features is adding a personal user bias into each local model to address heterogeneous behavior. Yet, according to the reported results, this approach produced only marginal improvements (up to $\sim 3\%$) in the overall prediction accuracy.

Two recently proposed FL4iMP solutions that address heterogeneity in user mobility more explicitly are *FR-HMP* and *FedGeo*. In *FR-HMP* [21], clients with similar local model parameters are grouped into clusters, enabling "similar" clients to learn from one another rather than from the whole population. Authors use the similarity between clients' model parameters as an *indirect measure* of the similarity between the mobility patterns of the corresponding users. *FedGeo* [22], in turn, introduces new components to the FL framework to mitigate the effects of client heterogeneity. One such component is a strategy to aggregate the currently learned global model into the selected clients' local models based on the layer-wise similarity between those models. This approach aims to mitigate the disparity among client models arising from the heterogeneity of user trajectories. It also employs entropy-based sampling to select clients to participate in FL based on the diversity of their mobility patterns, aiming at building more generalizable global models.

Although both studies acknowledge the challenge brought by heterogeneous user mobility patterns, they address heterogeneity from a model gradient perspective (i.e., two clients are diverse if their uploaded model parameters are dissimilar) or a data sampling perspective (i.e., two clients are dissimilar if they report diverse distributions of visited locations). Thus, pinpointing heterogeneity is tailored to the local model's capability (e.g., embedding, encoding) or to the clients' data quality in capturing diversity in mobility patterns. While such approaches report overall results under specific settings that are hard to generalize, they do not tackle heterogeneity from a human behavior point of view but rather from an indirect data and model perspective. Moreover, FR-HMP and FedGeo have been evaluated in very restricted scenarios regarding user populations (only 200 and 10 users, respectively).

A few other studies focus on varying the algorithm used to learn the local models. In [20], the authors adopt Recurrent Neural Networks (RNN), Long-Short-Term Memory (LSTM), and also Gated Recurrent Unit (GRU) [39] as alternative learning algorithms. This work shed some light on the discussion of which neural network to use in FL4iMP, with results

suggesting GRU as the best performer in most scenarios analyzed. In [19], in turn, the authors evaluate *FL-Flashback* and *GRU-Spatial*. The former is a federated version of *Flashback*, which was initially proposed as a centralized mobility prediction approach [9]. The core of the (FL-)*Flashback* algorithm is the use of spatial-temporal contexts: it searches for historical hidden states with context similar to the current one, computing the weighted average of these historical hidden states to feed a RNN-based model. Both temporal and spatial contexts are explored, with the weights decreasing as the time between visits increases and the distance between past and current locations increases. *GRU-Spatial*, in turn, keeps only the spatial perspective of the contexts, modeling temporal relations by the order of visits. In common, both studies give no attention to mobility heterogeneity, neither in the solution design nor the evaluation, and use the same strategy to aggregate local models, namely *FedAvg*.

Positioning: Prior work on FL4iMP focuses primarily on proposing new approaches, with evaluations that leave the FL4iMP research arena still very unsettled. Notably, no prior work, not even those that target user heterogeneity [11, 21, 22], report results separately for users with different mobility behaviors so as to allow an assessment of how FL4iMP performance may vary across different behaviors. Most studies also rely on LBSN datasets (mainly from the Foursquare location-sharing service), which, as will be argued in Section 3.2, tend to be very sparse and do not capture enough details of one's mobility. As such, results are hardly generalizable. Moreover, the differences in evaluation setup across those studies make it hard to reach a common knowledge on FL4iMP performance.

In contrast, we here offer, within a *unified evaluation environment*, a much broader investigation of how different factors related to human behavior, data, FL architecture and algorithms impact FL4iMP performance. Our evaluation considers metrics related to prediction accuracy, time efficiency as well as memory and energy usage. Moreover, unlike prior work, we offer insightful discussions on how FL4iMP performs for users with diverse mobility nature. To that end, we use two datasets with very different properties. We cluster users into mobility profiles and employ various mobility-related features to characterize such groups as well as the complete user population in each dataset. We use the characterization results to guide our evaluation and discussion of FL4iMP performance.

3 Key Factors Impacting FL4iMP

Based on how a typical FL4iMP solution works, we identify key factors that may impact its performance. We here group these factors into four categories, depending on whether they are related to: (1) *human* mobility behavior; (2) properties of the *data* capturing such behavior and used as input to the FL4iMP solution; (3) the learning *algorithms* employed to build both local and global models; and (4) the FL *architecture* itself.

3.1 Human factors

User mobility behavior is a significant element influencing mobility prediction accuracy and thus, FL4iMP performance, being the source of data for learning the prediction model. As such, naturally, factors related to user mobility patterns are of primary importance.

The mobility trajectory of a person can be split into two main components [25]. The *routine* comprises the locations the person often visits (e.g., home, work). In turn, the *novelty* component consists of places she rarely or for the first time visits, related thus to moments of exploration. It is well known that individual mobility is highly heterogeneous with respect to these two components, even if restricted to a common (small) area (e.g., a given city) [28, 30, 23].

More exploratory users are naturally less predictable, being thus more challenging for mobility prediction. While this is true for any prediction model, in FL4iMP, additional challenges may emerge from greater user heterogeneity. As further discussed in Section 3.3, users with very heterogeneous mobility patterns may generate local models (e.g., model parameters) that are very different from one another. This asymmetry can significantly challenge the aggregation algorithm, which might struggle to generate a global model suitable enough for all dissimilar clients.

This paper investigates how FL4iMP performs for users with heterogeneous mobility behavior. To that end, in Section 4, we employ a profiling approach [23, 24, 33] to group users according to their mobility patterns, and characterize the observed mobility patterns with respect to several commonly used features.

3.2 Data factors

Spatial-temporal datasets are of enormous importance to studying human mobility, consisting of *time-stamped and geo-referenced samples* of users' trajectories. Mobility datasets can be collected from a GPS-capable device or from a network, which gathers data when a mobile device interacts with it. For example, Telecom operator networks store Call Detailed Records (CDRs) containing information on generated events (e.g., calls, SMS) by mobile devices and the cell towers they were connected to and managed the event. In contrast, Location Based Social Networks (LBSNs), e.g.,

Foursquare, record the point of interest (POI) where the user willingly chose to share her location (i.e., check-in). To ensure unbiased and meaningful research on mobility prediction, datasets should capture:

1- Daily mobility behaviors: Datasets originated from different sources exhibit different characteristics, reflecting behaviors captured in the data. Such characteristics may impact how the user's true mobility is represented and, thus may affect FL4iMP performance. We argue that, for the sake of learning a proper mobility prediction model, the dataset must capture the *routine* component of the user mobility as much as possible. The novelty component, though reflecting important moments in the users' lives, has a low repetitive pattern, which adds noise to the learning process.

GPS datasets offer a detailed view of users' mobility, allowing a fine-grained investigation on mobility patterns. However, they are usually limited in the number of participants, time duration, and space coverage, raising concerns about the generality of the prediction results. In contrast, network-sourced (e.g., CDR and LBSN) datasets typically comprise larger populations, longer periods (e.g., months), and larger geographical areas (e.g., a city or region). Yet, their "interaction-demand" property raises concerns about how such actions relate to the routine component of a person's mobility.

One could argue that users more often place check-ins in newly visited places, while check-ins at one's home or workplace are rare [40]. Thus, certain LBSN datasets – e.g., from location-sharing service (e.g., *Foursquare*) – bring geographic information more often related to particular moments of leisure (e.g., bar, restaurant). As such, LBSN datasets often offer only a coarse view of user mobility, with low temporal and spatial granularity, which challenges its usage for learning mobility patterns.

In CDR datasets, in turn, the granularity will be dictated by how often the user places a call or an SMS. Nevertheless, with mobile devices becoming proxies for human presence and activity, CDRs are acknowledged by the research community as a meaningful tool to study human mobility [41], outlining many interesting properties regarding human mobility and activity patterns. Moreover, data completion strategies [42] have been commonly employed to reduce spatial-temporal sparsity in such datasets. Such observations makes CDRs a good candidate to investigate FL4iMP.

2- Diversity: Other important factors relate to the target population, spatial area, and the time period covered by the data. Naturally, more complex mobility patterns are expected over larger areas or regions with a greater diversity of locations. Similarly, the target area's social, economic, and geographical characteristics (e.g., distribution of residential and commercial areas, availability of transportation modes, etc) may also directly influence how people move around it. Such characteristics will be reflected in data properties such as the total number of unique locations or the number of (unique) locations in each user trajectory. Naturally, in scenarios of more complex and diverse mobility patterns, more data is required to learn accurate and generalizable models.

In sum, mobility datasets can vary significantly concerning properties that may impact FL4iMP. In this work, we aim to investigate some of them. While prior FL4iMP studies relied mainly on LBSN data [17, 18, 19, 20], we use two CDR datasets with distinct spatial-temporal properties, as will be discussed in Section 4.

3.3 Algorithm factors

The algorithms employed to learn the local models (at the clients) and to aggregate them into a global model (at the remote server) are the heart of the FL4iMP solution and thus play a crucial role in its performance. Regarding the (local) learning algorithms, the choice of which neural network to adopt reflects how the local algorithm captures and models the spatial-temporal relationships present in the input data, potentially impacting FL4iMP performance. A more sophisticated model may be able to capture these relationships more accurately, improving its predictive power. Yet, it may also challenge the clients, which, if constrained in resources (e.g., memory, energy), may struggle to process complex models efficiently.

Regarding the aggregation algorithm, one key factor relates to how the multiple local model weights are combined into a global model. The seminal *FedAvg* [12] algorithm simply averages the received local model weights. However, prior work has shown that FL performance may degrade in face of clients handling non-IID (independent and identically distributed) data [43]. This degradation occurs because the divergences in local models due to non-IID data slow down the convergence of the global model and worsen the learning performance.

Alternative aggregation methods have been designed to reduce the instabilities due to non-IID data [36, 22, 44] – i.e., the client drifts [37] – by avoiding the aggregation of dissimilar local model parameters of clients. For instance, *FedProx* reduces the distances between local and global models by incorporating a *Regularization term* (i.e., μ) in the local objective function that restricts the local updates, bringing the averaged model closer to the global optima. However, prior related work has not tackled heterogeneity from a human behavior perspective, but rather from the point of view of dissimilar model weights or data sampling distributions, i.e., model weights and data sampling are used as proxy metrics to infer heterogeneity in mobility patterns. By capturing human behavior representations (i.e., mobility profiles)

before the model training, we are able to study the impact of IID vs. non-IID data in user mobility prediction according to distinct mobility profiles. Additionally, the profile characterization brings interpretability to non-IID data, helping us understand the FL4iMP results – an investigation lacking in prior work. One aspect of particular interest is how fast the algorithm can generate a robust global model (i.e., converge). Considering that the aggregation algorithm generates the global model, and that aggregating similar model clients leads to faster global convergence, fewer communication rounds between server and local clients are needed, which can intuitively save resources. In our study, we evaluate two local learning algorithms and two aggregation methods, considering scenarios of both IID and non-IID clients. Our results will be discussed in Section 5.

3.4 Architecture factors

One key FL architectural factor relates to how users are distributed among the clients. As argued in Section 2.1, existing approaches use the *data silos* abstraction, aggregating multiple users' data per client. Such design abstraction may significantly impact user privacy and the performance of the FL4iMP solution. In fact, by leveraging data from multiple users into a local client, FL can train local models with superior generalization capabilities, potentially leading to more accurate predictions. This leads to a non-trivial trade-off between privacy and prediction accuracy. In Section 5, we assume such *data silos* architecture and assess the impact of increasing the number of FL4iMP clients. We also vary the distribution of the user population across the clients, creating scenarios of IID and non-IID *mobility behaviors*. This approach indirectly results in diverse distributions of mobility patterns (IID and non-IID) across the clients.

4 Dataset: Overview and Analysis

As argued in Section 3.2, our study relies on two CDR datasets with different spatial and temporal properties and diverse population characteristics. In this section, we first introduce the two datasets (Section 4.1) and how they were processed before being used (Section 4.2). These processed datasets exhibit a reasonable amount of data samples per user and consequently, across clients. This enables the investigation of heterogeneity stemming precisely from user mobility behaviors, rather than from data collection specificity (i.e., from the data sampling rate). Then, we investigate on an individual basis, the mobility behavior captured in both datasets. Our goal is not only to systematically asses whether the datasets exhibit heterogeneous properties (i.e., non-IID properties) but also to explain what aspects of users' mobility patterns make them dissimilar (Sections 4.3 and 4.4). To our knowledge, we are the first to *quantify* user heterogeneous mobility patterns before the FL training phase, i.e., directly from the raw data instead of from local or server model parameters (as in [21, 22, 36]) We are also the first to recognize that understanding the trade-offs dictating FL4iMP performance requires looking beyond aggregated results. We must assess how FL4iMP performs for users with diverse patterns separately. Next, we identify and characterize different mobility patterns in the two datasets. These results will be used to guide the evaluation of FL4iMP in Section 5.

4.1 CDR datasets

The first dataset, referred to as **Shanghai dataset**, was gathered by a major Telecom operator in the metropolitan area of Shanghai, China. It provides location data for 58,502 users, recorded at a frequency of one location per hour over ten days. This dataset is fully anonymized, with no user identity or corresponding cell tower infrastructure included. Unlike classical CDRs, the recorded location is not the coordinates of the Base Station (BS) to which the user was connected. Instead, it represents the centroid of a cell (in a grid representation) nearest to the BS the user was primarily connected to during each hour. In sum, the Shanghai dataset contains 10,396 distinct centroid coordinates and 9,135,780 records.

Our second dataset, referred to as **Shenzhen dataset**[2] [34], was collected by several major operators in Shenzhen (China). It includes the location data of 414,271 users collected by 1,090 Base Stations, resulting in 38,218,717 records. This dataset is also fully anonymized, with no user identities provided. Additionally, while records are associated with their collection times, all specific date information was removed, making it impossible to determine if records generated by any two users were collected on the same day.

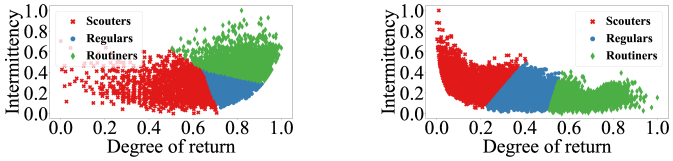
4.2 Data pre-processing

We pre-processed both datasets for fair comparison and unbiased analysis. First, to handle the restricted temporal information in the Shenzhen dataset, we assume the starting day d_1 of records is the same for all users in the collection. Time-ordered timestamps belonging to the same user are then associated to day d_x until a timestamp equal or earlier than the previous one is found, which triggers the start of a new day d_{x+1} for that user.

[2]https://people.cs.rutgers.edu/ dz220/data.html

Moreover, to mitigate the temporal gaps common to CDR datasets, we apply a data completion strategy to both datasets, as proposed in [42]. For this, records are conveniently added to complete or fill the gaps, as described next. Completion is done only for short intervals when the devices were not used, which allows for increasing the temporal resolution of the dataset without biasing results analysis. Specifically, we first identify three types of places for each user: *home*, *workplace A*, and *workplace B*. We define as *home* the most frequent daily location in which the user was from 2 a.m. to 6 a.m. Similarly, *workplace A* and *workplace B* are defined as the recurrent daily locations from 10 a.m. to 11 a.m. and 2 p.m. to 4 p.m., respectively. Once each place is identified, if a record is missing during each hourly interval during the mentioned periods, a new record is added with the most frequent location associated with each place type. By doing so, our completion approach aims at a temporal resolution of one hour during the three completed periods.

Next, we employ a 200mx200m square tessellation based on *OpenStreetMap* [45] (available in *Scikit-mobility* [46]) to map the location coordinates (latitude and longitude) into cells, we also replaced the original coordinates to the associated cell centroid coordinates. Ultimately, we conduct a data filtering that removes inactive users who could lead to poor mobility behavior identification and distorted evaluation results, given their low number of records. To this end, we first keep per user, one record for each distinct location visited in each 1h interval; then, we filter out users with less than 10 days and 120 records, as in [23] and [25].



(a) Shanghai. (b) Shenzhen.

Figure 1: Profiling of users' mobility behaviors per dataset.

Following the data processing, the Shanghai dataset comprises 58,471 users and 6,321 cells. Each user trajectory, on average, consists of 181 records (156 before the completion for the same users) spread over 10 days. In contrast, the processed Shenzhen dataset encompasses 42,266 users and 816 cells. On average, each user has 172 records (115 before completion) covering a period of 16.4 days. Thus, the two datasets exhibit very distinct properties concerning location diversity, temporal coverage, and user population. The Shanghai dataset has greater diversity in terms of unique locations and a larger user population. Also, mobility trajectories in Shanghai typically have more records. Yet, they tend cover shorter time periods, compared to those in Shenzhen.

Finally, both processed datasets have the same spatial resolution (i.e., 200mx200m cells) with the associated date and hour information. These similar spatial-temporal resolutions allow for a fair comparison and analysis between the two datasets, limiting the diversity of data sampling distribution to better focus on heterogeneity in mobility behaviors.

4.3 User mobility profiling

To capture the human factors discussed in Section 3.1, we perform a profiling of the users in each dataset. To this end, we apply the mobility modeling and profiling strategy described in [23, 33], yielding a powerful user characterization according to both routine and novelty components in their daily trajectories. The strategy consists in grouping users in the following three classes, whose temporal and spatial patterns will be analyzed in the next section:

Routiners: defines the set of users that have the *routine* as the most prevalent component in their daily mobility life, though also having moments of novelty-seeking; **Scouters:** are users prone to explore new areas, making the *novelty* the most prevalent component in their mobility pattern. Users in this profile very often break their routine while seeking for novelty, resulting in mobility patters that are more diverse and less regular; and **Regulars:** have a mobility pattern well balanced between routine and novelty moments, with no stronger predominance of one of them.

To create these profiles, we first employ the classification method described in [33], to categorize each location appearing in a user trajectory as either an exploration (i.e., a new or rarely visited location) or a return (i.e., a regularly visited location). The method first computes the importance of each location as the frequency of its appearance in the user's mobility trajectory. Once locations are classified, as in [23], two features per user are calculated: *Intermittency* and *Degree of Return*. *Intermittency* measures the frequency of shifts between explorations and returns, and *Degree of Return* indicates how often the user revisits a return-like location after an exploration-like one.

Computed values of *Intermittency* and *Degree of Return* are then scaled from 0 to 1 via a min-max normalization. Finally, we cluster the users using the two normalized features. As in [23, 33], we experimented with different clustering algorithms and variable number of clusters, using the *Silhouette* score [47] as metric of clustering quality. The best

results were obtained with the *k-means* algorithm, which split the users into the three aforementioned groups/profiles. Fig. 1 shows clustering results: each dot represents a user, colored according to the uncovered profile.

Insights: We observe the relationship between the profile definitions and the features (axes): *Scouters* and *Routiners* are on opposite sides and *Regulars* fall between them.

4.4 Data analysis

As presented in Section 4.2, the two datasets are very diverse with respect to user population, unique locations, and time periods. We now explore these differences by analyzing the user trajectories in each dataset using various mobility-related features. Our characterization considers all users and users in each identified profile, aiming to highlight their main differences and provide insights to help us understand the FL results presented in Section 5.

Our analysis relies on the following five features: (1) ***Regularity*** [26, 25] captures if an individual is prone to return to previously visited locations; (2) ***Stationarity*** [26, 25] quantifies if an individual stays continuously in the same location; (3) ***Diversity*** [25] measures how diverse an individual's visitation pattern is; (4) ***Entropy*** [31] provides a complementary view to Diversity, and builds upon the visitation frequency, the order in which the locations were visited, and the time spent at each location; and (5) ***Radius of gyration*** [28] measures whether a user has a confined mobility pattern (smaller values) concentrated in a specific area, or if she tends to travel longer distances (bigger values).

Regularity, *Stationarity* and *Diversity* are measured in the 0-1 scale: the closer the value is to 1, the more likely the individual is to exhibit the described behavior. Higher values of *Entropy* suggest harder to predict users' mobility. Trajectories with high entropy tend to have greater Diversity, lower Regularity, and Stationarity.

Table 4 presents the distribution of users across the three profiles for each dataset, along with the corresponding average numbers of records and unique locations per user trajectory. Figs. 2 and 3 show the distributions of the five aforementioned features' values for all users, as well as for users in each profile.

Shanghai dataset: As shown in the table, almost half of the population is dominated by users who often alternate between routine and novelty in their mobility patterns, i.e., *Regulars* (49%). The remaining ones are roughly balanced and split between *Scouters* (24%), who are more prone to novelty seeking, and *Routiners* (27%), who have a prevalent routine component. Moreover, even though users in all three profiles tend to have a similar number of records on average, *Scouters* have a much larger number of unique locations – more than twice that of *Routiners*.

Figs. 2a and 2b show that *Routiners* tend to have the highest values of *Regularity* and *Stationarity*, suggesting that these users are not only more prone to return to previously visited places (Fig. 2a) but also tend to stay longer in the same places (Fig. 2b). Consistently, they tend to have the lowest values of *Diversity* (Fig. 2c) and *Entropy* (Fig. 2d). Instead, *Scouters* have the highest values of *Diversity* and *Entropy* (and lowest *Regularity* and *Stationarity*), characterizing them as exploratory users with less predictable mobility trajectories. Finally, we notice that the Shanghai dataset is spatially wide, with some users having a *Radius of gyration* close to $45km$ (Fig. 2e).

Shenzhen dataset: Unlike the Shanghai dataset, the population in the Shenzhen dataset is dominated by users more prone to explore, i.e., Scouters (48%). Also, again, unlike in the Shanghai dataset, the number of records per user trajectory varies greatly across the profiles, with *Routiners* having up to 45% more records on average than *Scouters*. Regarding the number of unique locations, we find that, consistent with the observations made for the whole dataset (Section 4.2), i.e., user trajectories in the Shenzhen dataset are much less diversified. The distributions of *Regularity*, *Stationarity*, *Diversity*, and *Entropy* (Figs. 3a, 3b, 3c, 3d) show qualitatively similar patterns for the three profiles in the Shenzhen dataset as observed for the Shanghai one. Finally, compared to the Shanghai dataset, the Shenzhen dataset is much more spatially constrained (which explains the fewer number of unique locations), with *Radius of gyration* always lower than 20 km (Fig. 3e).

Insights: The mobility behaviors (captured by *Regularity*, *Stationarity*, *Diversity* and *Entropy*) of *Scouters*, *Regulars*, and *Routiners* are quite diverse, consistent with their definition, in both Shanghai and Shenzhen. However, the datasets highly differ in terms of the number of unique locations visited, the radius of gyration, and the number of records per trajectory. This discrepancy may facilitate prediction on the Shenzhen dataset, where user trajectories tend to have fewer distinct locations and be more confined, but with increased user history (i.e., data available for training the models). Regarding profiles, the entropy of *Scouters* (cf. *Routiners*) in both datasets is the highest (cf. lowest) among the three profiles, which will make individual mobility prediction more (cf. less) challenging.

Finally, the results demonstrate the datasets' diversity in spatial-temporal features and population heterogeneity, i.e., in terms of data and human factors (cf. Section 3). These assets provide the necessary foundation for studying the impact of mobility pattern heterogeneity on FL4iMP from a human behavior perspective. Besides, our analysis reveals the features that contribute to population mobility behavior heterogeneity, presenting three distinct profiles of users. This

clear differentiation among profiles and their features offers a robust basis for challenging mobility prediction and most importantly, for interpreting the impact of mobility heterogeneity (i.e., non-IID data) on FL4iMP results.

5 Evaluation

In this section, we dive into our evaluation of the impact of the various factors introduced in Section 3 on FL4iMP. Using the two datasets detailed in Section 4, we leverage the user profiles identified and characterized therein to capture human and data-related factors. We begin by outlining our experimental setup and explaining how we accounted for algorithm and architecture factors (Section 5.1). We then present our key findings, initially focusing on the human and data-related factors (Section 5.2), and subsequently on the factors related to the FL algorithms and architecture (Section 5.3).

5.1 Experimental methodology

FL architecture: To establish a unified environment for studying all identified factors, we used *Flower* [35], a state-of-the-art framework for FL. We configured *Flower* to distribute all users in the input dataset uniformly across all clients. We assume all clients participate in every round, leaving the analysis of alternative strategies for future work. This approach aims to minimize variability from diverse client selections, allowing us to focus on the number of clients (i.e., local models) as the primary architectural factor.

Local learning algorithms: Regarding the algorithm used to learn the local models, we chose to focus on two FL4iMP solutions based on more traditional and less costly neural network algorithms [48]: *FL-Flashback* and *GRU-Spatial* FL models [19]. These solutions rely on two distinct neural networks, namely RNN and GRU, respectively, as algorithm to learn the local models. We used the implementations of both *FL-Flashback* and *GRU-Spatial* developed in *Flower* and made available by the authors of [19].

Scenario	Acc@1	Acc@5	Energy (W)	VRAM (MB)	Train (s)	Test (s)	Total
1-Random Scouters (6K)	0.459 ± 0.00	0.651 ± 0.00	201.5 ± 1.35	6845 ± 3	15.5 ± 0.12	36.2 ± 0.17	516.6 ± 0.89 (08m 36s)
2-Random Regulars (6K)	0.542 ± 0.00	0.731 ± 0.00	200.8 ± 0.86	6439 ± 1	15.0 ± 0.04	**36.1 ± 0.07**	511.4 ± 0.70 (08m 31s)
3-Random Routiners (6K)	0.628 ± 0.00	0.780 ± 0.00	200.2 ± 0.76	6402 ± 1	14.9 ± 0.09	36.2 ± 0.04	510.6 ± 0.70 (08m 30s)
4-Super Scouters (6K)	0.343 ± 0.00	0.578 ± 0.00	202.3 ± 1.03	6558 ± 2	15.6 ± 0.00	36.2 ± 0.04	517.6 ± 0.43 (08m 37s)
5-Super Routiners (6K)	**0.805 ± 0.00**	**0.863 ± 0.00**	**196.5 ± 0.76**	**6226 ± 2**	**14.2 ± 0.04**	36.2 ± 0.04	**504.2 ± 0.66 (08m 24s)**
6-Equal mix (6K)	0.534 ± 0.00	0.710 ± 0.00	200.2 ± 0.65	6766 ± 3	15.4 ± 0.11	36.2 ± 0.12	516.0 ± 0.96 (08m 36s)
7-Original mix (6K)	0.542 ± 0.00	0.724 ± 0.00	199.9 ± 1.14	6479 ± 1	14.9 ± 0.12	36.1 ± 0.10	510.0 ± 1.11 (08m 30s)

(a) Shanghai dataset (GRU-Spatial, FedAvg, 6,000 users and 4 clients).

Scenario	Acc@1	Acc@5	Energy (W)	VRAM (MB)	Train (s)	Test (s)	Total
1-Random Scouters (6K)	0.504 ± 0.00	0.874 ± 0.00	181.6 ± 0.62	5663 ± 2	**10.0 ± 0.04**	**41.1 ± 0.09**	**511.2 ± 1.02 (08m 31s)**
2-Random Regulars (6K)	0.560 ± 0.00	0.906 ± 0.00	181.6 ± 0.82	5599 ± 2	10.5 ± 0.09	42.7 ± 0.06	532.2 ± 1.02 (08m 52s)
3-Random-Routiners (6K)	0.730 ± 0.00	0.973 ± 0.00	181.5 ± 0.59	5600 ± 1	14.9 ± 0.11	68.2 ± 0.25	831.2 ± 1.70 (13m 51s)
4-Super-Scouters (6K)	0.297 ± 0.00	0.745 ± 0.00	181.3 ± 0.92	5600 ± 1	11.2 ± 0.13	45.8 ± 0.24	570.6 ± 1.42 (09m 30s)
5-Super Routiners (6K)	**0.867 ± 0.00**	**0.988 ± 0.00**	181.3 ± 1.34	**5599 ± 1**	16.3 ± 0.07	64.8 ± 0.09	811.2 ± 1.02 (13m 31s)
6-Equal mix (6K)	0.619 ± 0.00	0.929 ± 0.00	180.9 ± 0.71	5665 ± 2	11.8 ± 0.12	49.9 ± 0.09	617.4 ± 0.89 (10m 17s)
7-Original mix (6K)	0.592 ± 0.00	0.917 ± 0.00	**180.8 ± 1.04**	5599 ± 1	11.2 ± 0.11	46.7 ± 0.18	579.6 ± 1.42 (09m 39s)

(b) Shenzhen dataset (GRU-Spatial, FedAvg, 6,000 users and 4 clients).

Table 1: Impact of the user mobility behaviors. Average results with 95% confidence intervals.

Aggregation algorithm: We consider the *FedAvg* [12] and the *FedProx* [36] aggregation strategies to build a global model, both available in *Flower*. Similarly to the *FedProx* discussion in [36], our analysis using the two datasets showed that the hyper-parameter $\mu = 1$ produces the best results, while smaller values closer to 0 yield similar results to *FedAvg*. Higher μ values (e.g., 1000) also produced results comparable to those for $\mu = 1$. Thus, we set μ to 1.

Parameterization: The hyper-parameters for the *FL-Flashback* and *GRU-Spatial* local models were set following their original configuration in [19]. Notably, the input trajectory length was set to 20. The number of epochs in the local model and the communication rounds between clients and remote servers were set to 10 (unless otherwise noted). Batch size, embedding size, and learning rate were set to 128, 10, and 0.001, respectively.

Hardware setup: Our experiments were executed in the following machine's configuration: NVIDIA GPU GeForce RTX 3090 (24,576 MB of VRAM), AMD Ryzen 5 5600X processor (6 cores / 12 threads) @ 3.7 GHz and 64 GB RAM (4x 16 GB DDR4) @ 2133 MT/s. As previously observed [19], GPU memory (VRAM) quickly becomes a bottleneck as the number of clients increases. Indeed, prior studies were restricted to at most 4 [19] or 10 [20] clients for their scenarios. We were able to reach as many as 17 clients in ours.

Evaluation scenarios: To isolate and assess the impact of the studied factors as much as possible, we build several different evaluation scenarios where all users either have the same profile (cf. Section 4.4), representing an IID data, or have mixed profiles, representing a non-IID case. We also varied the number of clients and the number of users per client. These scenarios will be introduced in the following sections, along with the corresponding results.

Training and test sets: For each scenario, defined by a set of users, we split the data into training and test sets as follows. We take the initial 80% of the records of all user trajectories as training set, which is used to learn the prediction model. The final 20% of the records of all trajectories are used as test set to evaluate it. We ran each scenario 5 times with random initial model weights.

Evaluation metrics: Our evaluation considers metrics of both prediction *effectiveness* and *efficiency*. To assess effectiveness, we follow prior work [11, 16, 17, 18, 19, 21, 22] and measure the number of times the correct location is among the top-k predicted locations (for k = 1 and 5), referred to as *accuracy in the top-k* or $Acc@k$. For efficiency, we measure the consumption of energy (in watts) and VRAM (in megabytes) used by the clients on the GPU, as well as the training, testing, and total execution times (in seconds). While the amount of data impacts all efficiency metrics, energy and VRAM usages are particularly sensitive to the complexity of users' mobility behaviors (i.e., of the processed data). We developed a Python script that measures the wattage and the used GPU VRAM every second. Efficiency metrics are aggregated across all clients, whereas $Acc@k$ is measured for each client and then aggregated. We report average results computed across 5 runs along with 95% confidence intervals for all metrics.

5.2 Human and data factors

Scenario	Acc@1	Acc@5	Energy (W)	VRAM (MB)	Train (s)	Test (s)	Total
8-Original mix (20K)	0.629 ± 0.00	0.916 ± 0.00	**206.3 ± 0.51**	**6600 ± 1**	**52.8 ± 0.10**	**128.3 ± 0.17**	**1811.2 ± 1.61 (30m 11s)**
9-Original mix (all)	**0.656 ± 0.00**	**0.955 ± 0.00**	209.2 ± 0.45	6687 ± 1	157.4 ± 0.16	374.5 ± 0.67	5319.4 ± 7.83 (1h 28m 39s)

Table 2: Impact of the user population size: average results with 95% confidence intervals. Scenarios described in Table 5. Shanghai dataset: (GRU Spatial, FedAvg, 4 clients Original mix (all) consists of 58K users).

We first focus on human and data factors, guided by the discussions in Section 4. To assess the impact of user mobility behavior on FL4iMP, we build various scenarios as described in Table 5, sampling users from both the Shanghai and Shenzhen datasets. To avoid overwhelming the discussion with excessive results, we concentrate on setups using *GRU-Spatial* as the local algorithm and *FedAvg* as the aggregation algorithm, with four clients. Yet, the qualitative conclusions remain similar across different configurations. Additionally, the input data is uniformly distributed across all four clients. Lets first consider the scenarios in rows 1-7 of Table 5, which, to control for population size, consists of samples of 6,000 users with diverse mobility patterns. For such scenarios, the average FL4iMP results are shown in Tables 1a and 1b for the Shanghai and Shenzhen datasets, respectively.

Effectiveness (column 2-3): Results vary greatly across the three user profiles (rows 1-3 in the tables) and even more for the two extreme scenarios of *4-Super Scouters (6K)* and *4-Super Routiners (6K)* (rows 4-5). In the Shanghai dataset, the average *Acc@1* reaches as high as 0.805 for *5-Super Routiners (6K)*, i.e., those with the easiest-to-predict mobility patterns, and as low as 0.345 for the hardest-to-predict *4-Super Scouters (6K)*. The gap is even larger in the Shenzhen dataset, due to an intricate combination of more extreme values of *Regularity*, *Stationarity*, and *Diversity* compared to those of Shanghai (see Figs. 2-3), and a smaller number of unique locations visited (see Table 4). As argued in Section 4.4, the Shenzhen dataset is much more spatially constrained, which naturally leads to simpler mobility patterns that are easier to learn and predict.

Indeed, the results for the two mixed scenarios (rows 6-7 of Tables 1a and 1b), which hardly reflect those for specific profiles (especially *Routiners* and *Scouters*), illustrate how average results for a heterogeneous user population, as reported in prior studies, can be misleading concerning the performance perceived by individual users (or user profiles). Such results vary somewhat depending on the distribution of profiles, being lower for cases where *Scouters*, who typically exhibit much more complex mobility patterns, are a smaller fraction (e.g., *6-Equal mix (6K)* in Shanghai).

Efficiency (column 4-8): Most results are quite similar across all scenarios for either dataset. Exceptions are the training, testing, and total execution times in the Shenzhen dataset, which are significantly longer for the *3-Random* and *5-Super Routiners (6K)* (row 3 and 5). This is due to the much larger number of records per trajectory for those users ($\approx 45\%$ higher in Table 4). In contrast, the number of records per trajectory in Shanghai is roughly the same for all profiles, leading to similar execution times in all phases. VRAM and energy usage are somewhat lower for *5-Super Routiners* in Shanghai, possibly due to a combination of profile simplicity in pattern prediction, fewer unique locations, and a roughly similar number of records per user trajectory (see Table 4).

Population size: Consider the last two scenarios of Table 5, where the profile distribution is maintained as in the original data, but the sample is increased to 20,000 users (*8-Original mix (20K)*) and the full dataset (*9-Original mix*

(all)). Tables 2 and 7 present the results for these scenarios for both datasets. These results should be compared to those in row 7 of Tables 1a and 1b (i.e., *7-Original mix (6k)*), for which the population is kept at 6,000 users.

Since the number of clients is fixed at 4, a larger population size results in more data (i.e., user trajectories) available for training the local models, thereby enhancing each client's training process. We observe that the execution times naturally increase with population size in both datasets as more data leads to more processing. Interestingly, the increases are only marginal regarding energy and VRAM consumption, suggesting the strong correlation of these two metrics with data complexity rather than data size. Indeed, since the profile distribution is the same in all scenarios of 2 and 7, as well as in the last scenario of 1a and 1b, the increase of population size (from 6k, to 20k, and to 58k (Shanghai) / 42k (Shenzhen)) does not significantly increase the complexity in learning and predicting. Still, learning benefits greatly from more data, justifying the slight increase in energy and VRAM. When comparing the two datasets, a much higher number of *unique* locations in Shanghai (see Table 4), naturally leads to more complex mobility patterns compared to Shenzhen, what is reflected in lower energy and VRAM consumption in this latter dataset.

Take-home message: FL4iMP performance is a multi-faceted problem where the final result depends, in non trivial ways, on the combination of human behavior and data properties. In particular it can be very sensitive to user mobility patterns, highlighting the need to take the inherent heterogeneity of human behavior into account when analyzing alternative solutions.

5.3 Impact of number of clients

Number of clients	Acc@1	Acc@5	Energy (W)	VRAM (MB)	Train (s)	Test (s)	Total
2 clients	**0.614 ± 0.00**	**0.864 ± 0.01**	**191.0 ± 0.88**	**3254 ± 1**	18.0 ± 0.07	39.1 ± 0.19	571.2 ± 2.03 (09m 31s)
4 clients	0.542 ± 0.00	0.724 ± 0.00	199.9 ± 1.14	6479 ± 1	14.9 ± 0.12	36.1 ± 0.10	510.0 ± 1.11 (08m 30s)
10 clients	0.391 ± 0.00	0.539 ± 0.00	205.4 ± 0.28	16161 ± 3	14.1 ± 0.04	**33.1 ± 0.07**	**473.0 ± 0.55 (07m 53s)**
15 clients	0.306 ± 0.00	0.441 ± 0.00	201.6 ± 0.82	24225 ± 7	**13.7 ± 0.14**	37.4 ± 0.09	511.2 ± 0.35 (08m 31s)

Table 3: Impact of the number of clients: average results with 95% confidence intervals. Shanghai dataset (Scenario *7-Original mix (6K)*, GRU-Spatial and FedAvg).

We now assess the impact of a key architectural parameter, namely, the number of clients. To that end, we keep the population size fixed at 6,000 users, i.e., *7-Original mix (6K)* scenario (see Table 5), and use *GRU-Spatial* and *FedAvg* as the local learning algorithm and aggregation strategy, respectively. We varied the number of clients from 2 to the maximum possible in our setup (before running out of VRAM), which was 15 for the Shanghai and 17 for the Shenzhen datasets, respectively. Results are shown in Tables 3 and 8 (results for 4 clients are the same as those in Table 1 for the same scenario).

Since the population is kept fixed, increasing the number of clients results in a reduction of the amount of data available for learning the local models. As shown, accuracy drops significantly as the number of clients increases, especially for the Shanghai dataset, which has trajectories with more distinct locations. The reduced data available to each client challenges the effective learning of the more complex mobility patterns.

Resource consumption naturally increases with the number of clients due to contention on the shared hardware. Conversely, execution time, especially for local model training, decreases as the amount of data decreases. Interestingly, we see a change of pattern from 10 to 15 clients. This occurs because, in this case, the number of CPU threads available (12) is exceeded. The CPU is responsible for loading the data into the GPU. Thus, resource contention arises as the number of CPU threads becomes scarce to feed all clients, leading to extra delays. Energy consumption, in turn, reduces as the GPU may occasionally become idle, waiting for the CPU. Such idle periods also contribute to increasing execution times.

5.4 Impact of the local algorithm

All the results discussed so far were obtained by using *GRU-Spatial* as the learning algorithm at the clients. We now focus on this specific factor and compare *GRU-Spatial* against *FL-Flashback*. We consider a population of 6,000 users and 4 clients and, once again, use *FedAvg* to aggregate learned local models into a global model.

As discussed in Section 2.2, these algorithms' accuracy and training times were analyzed using LBSN datasets in [19]. The reported results suggest that in terms of accuracy, there is no clear winner. Yet, unlike *GRU-Spatial*, *FL-Flashback* captures both temporal and spatial contexts, which requires much longer training times.

Our results are shown in Table 9. For ease of comparison, the GRU-Spatial results are repeated from Table 1, which correspond to the same scenarios. In terms of accuracy, *GRU-Spatial* consistently outperforms *FL-Flashback* in almost all cases. This superior performance can be attributed to the gating mechanisms within every GRU cell, which allow

the model to learn which data in a sequence is important to retain and which data should be discarded or ignored [49]. However, the performance gap between the algorithms varies depending on the complexity of the mobility profiles in the input data. For example, for the *4-Super Scouters (6K)* scenario, which consists of users with the harder-to-predict profile, the improvements in *Acc@1* are around 20% and 29% for the Shanghai and Shenzhen datasets, respectively. For the simpler *5-Super Routiners (6K)* scenario, *GRU-Spatial* yields a 94% improvement in *Acc@1* over *FL-Flashback* in Shanghai. The only case where both algorithms deliver similar performance (with a slight advantage to *FL-Flashback*) is the *5-Super Routiners (6K)* in the Shenzhen dataset. Users in this scenario exhibit very simple (i.e., easy-to-predict) mobility trajectories, which are much easier than the same profile in Shanghai. Indeed, the *5-Super Routiners (6K)* scenario in Shenzhen has only 540 distinct locations, compared to 3,919 in Shanghai. This suggests that the Shenzhen sample has much more redundancy, making it easier for both algorithms to perform reasonably well.

Regarding training times, our results agree with those reported in [19]: *GRU-Spatial* has much faster training times, being also 5-times smaller (i.e., having about 5-times less trainable parameters) than Flashback model. Interestingly, despite having more parameters, we observe that *FL-Flashback* consistently consumes less energy and VRAM compared to *GRU-Spatial*. The reason for this lower resource usage might be due to the differences in how the two models handle computational resources. *Flashback*, although having more parameters, appears to be designed in a way that is more computationally efficient, possibly optimizing its operations to reduce resource usage. In contrast, GRU-Spatial, with its efficient architecture and specialized gating mechanisms, lead to higher computational demands to manage and process the data. These gating mechanisms allow GRU-Spatial to effectively learn which parts of the input data are important, improving prediction accuracy. However, this process is computationally intensive, leading to higher energy consumption and VRAM usage during training and inference. Thus, while GRU-Spatial is more accurate and fast, the trade-off is higher in resource consumption compared to Flashback.

Take-home message: Our results on the impact of the number of clients and the learning algorithm agree with prior work on a qualitative level. Yet, we here show, once again, the need to investigate FL4iMP performance separately for diverse mobility profiles, as performance gaps exhibit striking differences. Whereas for some mobility profiles, changing the learning algorithm has a strong impact (e.g., *5-Super Routiners (6K)* in Shanghai), in scenarios of more complex patterns, the impact is more modest (e.g., *4-Super Scouters (6K)* in both datasets). In cases of very simple mobility patterns, the impact may be marginal (e.g., *Super Routiners* in Shenzhen). Yet, the resource usage results of the two learning algorithms suggest that there is a balance between achieving high prediction accuracy and the amount of computational resources required, which is an important consideration for practical applications.

5.5 Impact of data distribution across clients

We here evaluate IID vs. non-IID users distributions across a variety of scenarios. Table 6 presents each scenario evaluated. Each scenario contains 6,000 users, distributed across 3 clients. The scenario *10-Mixed IID (6K)* contains a mix of profiled users (about 30%-37% *Scouters* and 32%-35% *Routiners*) on each client, representing the traditional approach of combining users with distinct patterns in the same client [19, 20]. In the *11-Mixed non-IID (6K)* scenario, the **exact same users** of the previous scenario are now divided between clients given their mobility profile, where: Client 1 has 2,000 (*Super*) *Scouters* with the highest values of entropy; Client 2 has 2,000 Random Regulars randomly selected from the 6,000 in table 5; and Client 3 has 2,000 (*Super*) *Routiners* with the lowest values of entropy. Next, the scenarios *12-Only Scouters IID (6K)* and *13-Only Routiners IID (6K)*, are IID scenarios with only Scouters or Routiners, divided by their entropy. Table 10 reports accuracy results on the scenarios *10-Mixed IID (6K)* and *11-Mixed non-IID (6K)*, with GRU-Spatial and both FedAvg and FedProx.

Combining users with different mobility profiles (*10-Mixed IID (6K)*) leads to misleading accuracy results that are not aligned with the entropy results of profiles shown in Figs. 2 and 3 for the two datasets. *Super Scouters*, for instance, reach the highest entropy in both datasets, indicating how troublesome their predictability is, i.e., how difficult it is to correctly predict their location visits compared to *Super Routiners* users. However, the *10-Mixed-IID (6K)* scenario hides the natural prediction uncertainty of *Super Scouters*'s mobility by reporting average *Acc@1* values that are higher than 0.572 in both datasets and equal to mixed users from *Super Routiners* or *Regulars* profiles. This occurs because combining users and simply reporting the resulting average neglects the hard/easy predictability of each user. In the *11-Mixed non-IID (6K)* scenario, we notice the discrepancy between the hardest to predict *Super Scouters (2K)* users of Client 1 (i.e., with *Acc@1* of 0.221) and the easiest to predict *Super Routiners (2K)* users of client 3 (i.e., with *Acc@1* of 0.763). This discrepancy ultimately reduces the resulting average *Acc@1* to 0.519, highlighting the performance obfuscation across clients of the *10-Mixed IID (6K)* scenario. Furthermore, we notice that *FedProx* produced slightly better results for most non-IID clients than *FedAvg*, due to its regularization described in Section 3.3.

Next, we separately compare scenarios with either *Super Scouters* or *Super Routiners* users. Our goal is to asses if combining similar mobility patterns across clients can improve accuracy or convergence speed. Fig. 4 presents how such *Super* users perform in scenarios having mixed profiled users (i.e., Clients 1 and 3 of *11-Mixed non-IID (6K)*

scenario) compared to scenarios only having similar profiles (i.e., Client 1 in *12-Only Scouters IID (6K)* or in *13-Only Routiners IID (6K)*). As depicted, separating clients by profile produce better results, i.e., faster convergence, specially for clients harder to predict, i.e., *Scouters*. In results not reported here for lack of space, we observe *Acc@1* values of *Light, Moderate*, and *Super Scouters* in the *12-Only Scouters IID (6K)* scenario that are higher than the one of *Super Scouters* in the (*10-Mixed non-IID (6K)*) scenario. This results correlates with the aggregation algorithm, that can generate a global model more helpful for the clients having closer mobility patterns.

Take-home message: Mixed users distribution across clients significantly impact the performance of FL4iMP models, even when the same number of users per client is considered. As shown, mixing users of dissimilar mobility patterns in the same clients obscure performance differences that emerge for users with diverse mobility profiles (i.e., the *Mixed IID* case), and separating them by their patterns (i.e., the *Mixed non-IID* case) highlights the hidden heterogeneity. Finally, combining users with similar behaviours (i.e., the *Only* case) facilitates the aggregation process, leading to better results, specially for harder to predict users.

6 Conclusions

In this work, we explored the non-triviality of FL4iMP and how it is impacted by a range of factors. While FL4iMP literature primarily focus on aggregated results and tackle mobility heterogeneity indirectly from a data and model perspective, we offered a broader investigation on how FL4iMP performs on users with diverse mobility profiles, tackling the heterogeneity problem directly from a human mobility behavior perspective. Our results showed that FL4iMP performance is a multi-faceted problem that depends on the combination of human behavior and data properties, being very sensitive to diversity in mobility pattern, and requiring from learning algorithms' design, a balance between accuracy and computational resources. We highlight the fact that disregarding the pinpointed factors impacting FL4iMP, leads to poor and misleading performance results, obfuscating differences that emerge for users with different mobility patterns. Our findings recommend combining users with similar behaviors to easier aggregation process, improve performance, and resources usability. Future work include new and adaptive FL4iMP models, e.g.: solutions that adopt complex models for hard-to-predict users and a simplistic models for the simplest ones, coupled with an architecture with the number of global servers equalling the profiles.

References

[1] David E Boyce and Huw CWL Williams. *Forecasting urban travel: Past, present and future*. Edward Elgar Publishing, 2015.

[2] Khong-Lim Yap, Yung-Wey Chong, and Weixia Liu. Enhanced handover mechanism using mobility prediction in wireless networks. *PloS one*, 15(1):e0227982, 2020.

[3] Hamada S Badr, Hongru Du, Maximilian Marshall, Ensheng Dong, Marietta M Squire, and Lauren M Gardner. Association between mobility patterns and COVID-19 transmission in the USA: a mathematical modelling study. *The Lancet Infectious Diseases*, 20(11):1247–1254, 2020.

[4] Yacine Belal, Sonia Ben Mokhtar, Hamed Haddadi, Jaron Wang, and Afra Mashhadi. Survey of federated learning models for spatial-temporal mobility applications. *arXiv preprint arXiv:2305.05257*, 2023.

[5] Sébastien Gambs, Marc-Olivier Killijian, and Miguel Núñez del Prado Cortez. Next place prediction using mobility markov chains. In *Proceedings of the first workshop on measurement, privacy, and mobility*, pages 1–6, 2012.

[6] Wesley Mathew, Ruben Raposo, and Bruno Martins. Predicting future locations with hidden Markov models. In *Proceedings of the 2012 ACM conference on ubiquitous computing*, pages 911–918, 2012.

[7] Andrea Cuttone, Sune Lehmann, and Marta C González. Understanding predictability and exploration in human mobility. *EPJ Data Science*, 7:1–17, 2018.

[8] Jie Feng, Yong Li, Chao Zhang, Funing Sun, Fanchao Meng, Ang Guo, and Depeng Jin. Deepmove: Predicting human mobility with attentional recurrent networks. In *Proceedings of the 2018 world wide web conference*, pages 1459–1468, 2018.

[9] Dingqi Yang, Benjamin Fankhauser, Paolo Rosso, and Philippe Cudre-Mauroux. Location prediction over sparse user mobility traces using rnns. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pages 2184–2190, 2020.

[10] Zongyuan Huang, Shengyuan Xu, Menghan Wang, Hansi Wu, Yanyan Xu, and Yaohui Jin. Human mobility prediction with causal and spatial-constrained multi-task network. *EPJ Data Science*, 13(1):22, 2024.

[11] Jie Feng, Can Rong, Funing Sun, Diansheng Guo, and Yong Li. PMF: A privacy-preserving human mobility prediction framework via federated learning. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 4(1):1–21, 2020.

[12] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[13] Tuo Zhang, Lei Gao, Chaoyang He, Mi Zhang, Bhaskar Krishnamachari, and A Salman Avestimehr. Federated learning for the internet of things: Applications, challenges, and opportunities. *IEEE Internet of Things Magazine*, 5(1):24–29, 2022.

[14] Junghye Lee, Jimeng Sun, Fei Wang, Shuang Wang, Chi-Hyuck Jun, Xiaoqian Jiang, et al. Privacy-preserving patient similarity learning in a federated environment: development and analysis. *JMIR medical informatics*, 6(2):e7744, 2018.

[15] Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*, 2018.

[16] Zipei Fan, Xuan Song, Renhe Jiang, Quanjun Chen, and Ryosuke Shibasaki. Decentralized attention-based personalized human mobility prediction. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(4):1–26, 2020.

[17] Anliang Li, Shuang Wang, Wenzhu Li, Shengnan Liu, and Siyuan Zhang. Predicting human mobility with federated learning. In *Proceedings of the 28th International Conference on Advances in Geographic Information Systems*, pages 441–444, 2020.

[18] Shuang Wang, Bowei Wang, Shuai Yao, Jiangqin Qu, and Yuezheng Pan. Location prediction with personalized federated learning. *Soft Computing*, pages 1–12, 2022.

[19] Castro Elizondo Jose Ezequiel, Martin Gjoreski, and Marc Langheinrich. Federated Learning for Privacy-Aware Human Mobility Modeling. *Frontiers in Artificial Intelligence*, 5, 2022.

[20] Vlad-Alexandru Proteasa, Radu-Ioan Ciobanu, Ciprian Dobre, and Radu-Corneliu Marin. Federated Learning for Human Mobility. In *2023 19th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, pages 780–785. IEEE, 2023.

[21] Xiao Zhang, Qilin Wang, Ziming Ye, Haochao Ying, and Dongxiao Yu. Federated representation learning with data heterogeneity for human mobility prediction. *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[22] Chung Park, Taekyoon Choi, Taesan Kim, Mincheol Cho, Junui Hong, Minsung Choi, and Jaegul Choo. FedGeo: Privacy-Preserving User Next Location Prediction with Federated Learning. In *Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems*, pages 1–10, 2023.

[23] Licia Amichi, Aline Carneiro Viana, Mark Crovella, and Antonio AF Loureiro. Understanding individuals' proclivity for novelty seeking. In *Proceedings of the 28th international conference on advances in geographic information systems*, pages 314–324, 2020.

[24] Licia Amichi, Aline Carneiro Viana, Mark Crovella, and Antonio AF Loureiro. From movement purpose to perceptive spatial mobility prediction. In *Proceedings of the 29th International Conference on Advances in Geographic Information Systems*, pages 500–511, 2021.

[25] Douglas do Couto Teixeira, Jussara M Almeida, and Aline Carneiro Viana. On estimating the predictability of human mobility: the role of routine. *EPJ Data Science*, 10(1):49, 2021.

[26] Douglas Do Couto Teixeira, Aline Carneiro Viana, Mário S Alvim, and Jussara M Almeida. Deciphering predictability limits in human mobility. In *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 52–61, 2019.

[27] Douglas Do Couto Teixeira, Aline Carneiro Viana, Jussara M Almeida, and Mrio S Alvim. The impact of stationarity, regularity, and context on the predictability of individual human mobility. *ACM Transactions on Spatial Algorithms and Systems*, 7(4):1–24, 2021.

[28] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *nature*, 453(7196):779–782, 2008.

[29] Gavin Smith, Romain Wieser, James Goulding, and Duncan Barrack. A refined limit on the predictability of human mobility. In *2014 IEEE international conference on pervasive computing and communications (PerCom)*, pages 88–94. IEEE, 2014.

[30] Eduardo Mucceli, Aline Carneiro Viana, Carlos Sarraute, Jorge Brea, and José Ignacio Alvarez-Hamelin. On the Regularity of Human Mobility. *Pervasive and Mobile Computing*, 2016.

[31] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-László Barabási. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, 2010.

[32] Luca Canzian and Mirco Musolesi. Trajectories of depression: unobtrusive monitoring of depressive states by means of smartphone mobility traces analysis. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, pages 1293–1304, 2015.

[33] Licia Amichi, Aline Viana Carneiro, Mark Crovella, and Antonio Loureiro. Revealing an inherently limiting factor in human mobility prediction. *IEEE Transactions on Emerging Topics in Computing*, 2022.

[34] Desheng Zhang, Juanjuan Zhao, Fan Zhang, and Tian He. UrbanCPS: a cyber-physical system based on multi-source big infrastructure data for heterogeneous model integration. In *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems*, pages 238–247, 2015.

[35] Daniel J Beutel, Taner Topal, Akhil Mathur, Xinchi Qiu, Javier Fernandez-Marques, Yan Gao, Lorenzo Sani, Kwing Hei Li, Titouan Parcollet, Pedro Porto Buarque de Gusmão, et al. Flower: A Friendly Federated Learning Research Framework. *arXiv preprint arXiv:2007.14390*, 2020.

[36] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

[37] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated Learning on Non-IID Data Silos: An Experimental Study. In *2022 IEEE 38th international conference on data engineering (ICDE)*, pages 965–978. IEEE, 2022.

[38] Craig Stedman. What are data silos and what problems do they cause? `https://www.techtarget.com/searchdatamanagement/definition/data-silo`, 2021. [Accessed on 07-May-2024].

[39] Charu C. Aggarwal. *Neural Networks and Deep Learning: A Textbook*. Springer Publishing Company, Incorporated, 1st edition, 2018.

[40] Thiago H. Silva, Aline Carneiro Viana, Fabrício Benevenuto, Leandro Villas, Juliana Salles, Antonio A. F. Loureiro, and Daniele Quercia. Urban Computing Leveraging Location-Based Social Network Data: a Survey. *ACM Computing Surveys*, March 2019.

[41] Shan Jiang, Gaston A Fiore, Yingxiang Yang, Joseph Ferreira Jr, Emilio Frazzoli, and Marta C González. A review of urban computing for mobile phone traces: current methods, challenges and opportunities. In *Proceedings of the 2nd ACM SIGKDD international workshop on Urban Computing*, pages 1–9, 2013.

[42] Guangshuo Chen, Aline Carneiro Viana, Marco Fiore, and Carlos Sarraute. Complete trajectory reconstruction from sparse mobile phone data. *EPJ Data Science*, 8(1):1–24, 2019.

[43] Hangyu Zhu, Jinjin Xu, Shiqing Liu, and Yaochu Jin. Federated learning on non-IID data: A survey. *Neurocomputing*, 465:371–390, 2021.

[44] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE transactions on neural networks and learning systems*, 32(8):3710–3722, 2020.

[45] OpenStreetMap contributors. Planet dump retrieved from https://planet.osm.org . `https://www.openstreetmap.org`, 2017.

[46] Luca Pappalardo, Filippo Simini, Gianni Barlacchi, and Roberto Pellungrini. scikit-mobility: A Python Library for the Analysis, Generation, and Risk Assessment of Mobility Data. *Journal of Statistical Software*, 103(1):1–38, 2022.

[47] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

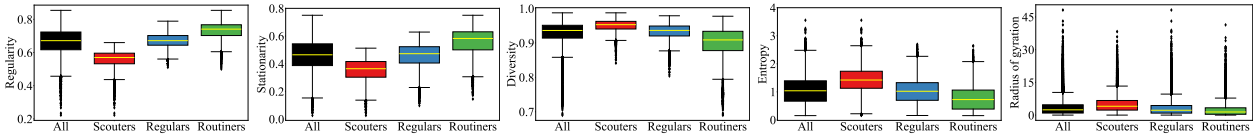
[48] Washington Cunha, Vítor Mangaravite, Christian Gomes, Sérgio Canuto, Elaine Resende, Cecilia Nascimento, Felipe Viegas, Celso França, Wellington Santos Martins, Jussara M Almeida, et al. On the cost-effectiveness of neural and non-neural approaches and representations for text classification: A comprehensive comparative study. *Information Processing & Management*, 58(3):102481, 2021.

[49] Massimiliano Luca, Gianni Barlacchi, Bruno Lepri, and Luca Pappalardo. A survey on deep learning for human mobility. *ACM Computing Surveys (CSUR)*, 55(1):1–44, 2021.

A Appendix

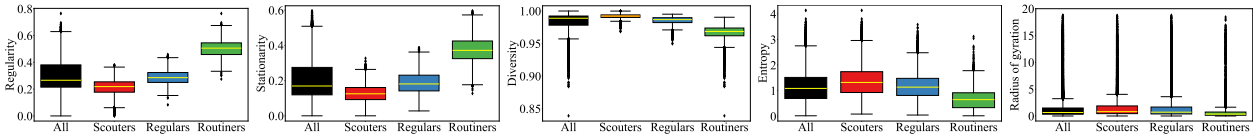
	Shanghai dataset			Shenzhen dataset		
Mobility profile	**# users**	**# records**	**# unique locations**	**# users**	**#records**	**# unique locations**
Scouters	13,728 (24%)	179.23 ± 0.07	**19.21 ± 0.14**	**20,447 (48%)**	153.95 ± 0.45	**8.89 ± 0.10**
Regulars	**28,679 (49%)**	181.36 ± 0.05	11.38 ± 0.07	11,716 (28%)	159.39 ± 1.12	8.17 ± 0.14
Routiners	16,064 (27%)	**182.84 ± 0.07**	8.72 ± 0.09	10,103 (24%)	**223.33 ± 1.23**	5.56 ± 0.09

Table 4: Average number of users, records, and unique locations per profile for the two datasets.



(a) Regularity. (b) Stationarity. (c) Diversity. (d) Entropy. (e) Radius of gyration.

Figure 2: Mobility behavior features' analysis for the Shanghai dataset.



(a) Regularity. (b) Stationarity. (c) Diversity. (d) Entropy. (e) Radius of gyration.

Figure 3: Mobility behavior features' analysis for the Shenzhen dataset.

Scenario	Description
1-Random *Scouters (6K)*	6,000 users randomly selected from all Scouters.
2-Random *Regulars (6K)*	6,000 users randomly selected from all Regulars.
3-Random *Routiners (6K)*	6,000 users randomly selected from all Routiners.
4-*Super-Scouters (6K)*	6,000 users with the highest values of entropy, being those with the hardest to predict mobility patterns.
5-*Super-Routiners (6K)*	6,000 users with lowest entropy (easiest to predict).
6-*Equal mix (6K)*	6,000 users, with 2,000 randomly sampled from each profile.
7-*Original mix (6K)*	6,000 users randomly sampled with the distribution of user profiles kept the same as in the original data.
8-*Original mix (20K)*	20K users randomly sampled with the distribution of user profiles kept the same as in the original data.
9-*Original mix (all)*	Full original dataset.

Table 5: Scenarios for human and data factors' analysis. Profile distributions of Scenarios 7-9: For Shanghai: 24% Scouters, 49% Regulars and 27% Routiners. For Shenzhen: 48% Scouters, 28% Regulars and 24% Routiners.

Scenario	Client 1 (2K)	Client 2 (2K)	Client 3 (2K)
10-Mixed IID (6K)	Mixed (Super Scts./Random Regs./Super Rots.)	Mixed (Super Scts./Random Regs./Super Rots.)	Mixed (Super Scts./Random Regs./Super Rots.)
11-Mixed non-IID (6K)	Super Scouters (2K) (top 2K in entropy)	Random Regulars (2K Regulars)	Super Routiners (2K) (bottom 2K in entropy)
12-Only Scouters IID (6K)	Super Scouters (top 2K in entropy)	Moderate Scouters (2K) (top 2K to 4K in entropy)	Light Scouters (top 4K to 6K in entropy)
13-Only Routiners IID (6K)	Super Routiners (bottom 2K in entropy)	Moderate Routiners (bottom 2K to 4K in entropy)	Light Routiners (bottom 4K to 6K in entropy)

Table 6: Scenarios to analyze IID and non-IID data across clients.

Scenario	Acc@1	Acc@5	Energy (W)	VRAM (MB)	Train (s)	Test (s)	Total
8-Original mix (20K)	0.601 ± 0.00	0.939 ± 0.00	**185.0 ± 0.66**	**5609 ± 1**	**37.8 ± 0.13**	**162.6 ± 0.16**	**2004.6 ± 1.89 (33m 24s)**
9-Original mix (all)	**0.607 ± 0.00**	**0.942 ± 0.00**	185.0 ± 0.67	5672 ± 1	80.9 ± 0.23	347.7 ± 0.87	4286.4 ± 8.03 (1h 11m 26s)

Table 7: Impact of the user population size: average results with 95% confidence intervals. Scenarios described in Table 5. Shenzhen dataset (GRU Spatial, FedAvg, 4 clients, Original mix (all) consists of 42K users).

Number of clients	Acc@1	Acc@5	Energy (W)	VRAM (MB)	Train (s)	Test (s)	Total
2 clients	**0.593 ± 0.00**	**0.935 ± 0.00**	**172.3 ± 1.11**	**2812 ± 1**	16.8 ± 0.12	49.7 ± 0.10	665.4 ± 1.80 (11m 05s)
4 clients	0.592 ± 0.00	0.917 ± 0.00	180.8 ± 1.04	5599 ± 1	11.2 ± 0.11	46.7 ± 0.18	579.6 ± 1.42 (09m 39s)
10 clients	0.584 ± 0.00	0.843 ± 0.00	183.7 ± 0.83	13964 ± 4	**9.6 ± 0.04**	46.4 ± 0.04	559.8 ± 0.35 (09m 19s)
15 clients	0.570 ± 0.00	0.813 ± 0.00	182.5 ± 1.07	20931 ± 3	10.6 ± 0.09	**45.0 ± 0.07**	**555.2 ± 0.66 (09m 15s)**
17 clients	0.563 ± 0.00	0.802 ± 0.00	181.2 ± 0.44	23714 ± 1	10.9 ± 0.14	46.1 ± 0.14	571.2 ± 0.66 (09m 31s)

Table 8: Impact of the number of clients: average results with 95% confidence intervals. Shenzhen dataset (Scenario *7-Original mix (6K)*, GRU-Spatial and FedAvg).

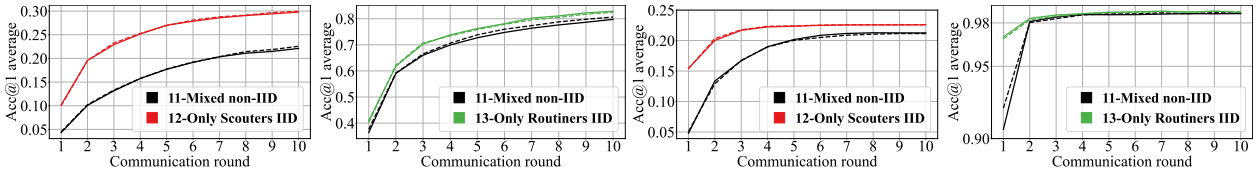
Scenario	Local algorithm	Acc@1	Acc@5	Energy (W)	VRAM (MB)	Train (s)	Test (s)	Total
4-Super Scouters (6K)	**FL-Flashback**	0.266 ± 0.00	0.453 ± 0.00	**178.5 ± 0.88**	**6249 ± 1**	76.0 ± 0.37	37.6 ± 0.09	1136.0 ± 4.00 (18m 56s)
	GRU-Spatial	**0.343 ± 0.00**	**0.578 ± 0.00**	202.3 ± 1.03	6558 ± 2	**15.6 ± 0.00**	**36.2 ± 0.04**	**517.6 ± 0.43 (08m 37s)**
5-Super Routiners (6K)	**FL-Flashback**	0.415 ± 0.00	0.607 ± 0.00	**176.3 ± 1.30**	**5991 ± 2**	76.3 ± 0.16	37.4 ± 0.12	1137.0 ± 1.47 (18m 57s)
	GRU-Spatial	**0.805 ± 0.00**	**0.863 ± 0.00**	196.5 ± 0.76	6226 ± 2	**14.2 ± 0.04**	**36.2 ± 0.04**	**504.2 ± 0.66 (08m 24s)**

(a) Shanghai dataset (FedAvg and 4 clients).

Scenario	Local algorithm	Acc@1	Acc@5	Energy (W)	VRAM (MB)	Train (s)	Test (s)	Total
4-Super Scouters (6K)	**FL-Flashback**	0.246 ± 0.00	0.564 ± 0.00	**172.2 ± 1.04**	**5506 ± 0**	69.0 ± 0.40	47.8 ± 0.13	1167.6 ± 4.06 (19m 27s)
	GRU-Spatial	**0.297 ± 0.00**	**0.745 ± 0.00**	181.3 ± 0.92	5600 ± 1	**11.2 ± 0.13**	**45.8 ± 0.24**	**570.6 ± 1.42 (09m 30s)**
5-Super Routiners (6K)	**FL-Flashback**	**0.872 ± 0.00**	0.951 ± 0.00	**171.4 ± 0.50**	**5505 ± 1**	100.1 ± 0.38	67.6 ± 0.04	1676.8 ± 4.09 (27m 56s)
	GRU-Spatial	0.867 ± 0.00	**0.988 ± 0.00**	181.3 ± 1.34	5599 ± 1	**16.3 ± 0.07**	**64.8 ± 0.09**	**811.2 ± 1.02 (13m 31s)**

(b) Shenzhen dataset (FedAvg and 4 clients).

Table 9: Impact of the local algorithm: average results with 95% confidence intervals.



(a) Super Scouters (2K), Shanghai. (b) Super Routiners (2K), Shanghai. (c) Super Scouters (2K), Shenzhen. (d) Super Routiners (2K), Shenzhen.

Figure 4: Impact of data distribution on *Acc@1* and convergence in scenarios of Table 6. FedAvg (Solid line). FedProx (Dashed).

Dataset	Shanghai				Shenzhen			
Scenario	10-Mixed IID		11-Mixed non-IID		10-Mixed IID		11-Mixed non-IID	
Client	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5
1	0.586 ± 0.01	**0.764 ± 0.00**	0.221 ± 0.01	0.443 ± 0.01	0.631 ± 0.00	**0.865 ± 0.00**	0.213 ± 0.00	0.579 ± 0.00
2	0.572 ± 0.00	0.753 ± 0.01	0.540 ± 0.01	0.764 ± 0.01	**0.636 ± 0.00**	0.861 ± 0.00	0.549 ± 0.00	0.905 ± 0.00
3	**0.591 ± 0.00**	0.763 ± 0.00	**0.797 ± 0.01**	**0.886 ± 0.01**	0.603 ± 0.00	0.836 ± 0.00	**0.986 ± 0.00**	**0.999 ± 0.00**
Avg.	0.583 ± 0.00	0.760 ± 0.00	0.519 ± 0.01	0.698 ± 0.01	0.623 ± 0.00	0.854 ± 0.00	0.583 ± 0.00	0.828 ± 0.00

(a) FedAvg, GRU-Spatial, 6,000 users and 4 clients.

Dataset	Shanghai				Shenzhen			
Scenario	10-Mixed IID		11-Mixed non-IID		10-Mixed IID		11-Mixed non-IID	
Client	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5
1	0.583 ± 0.01	**0.764 ± 0.00**	0.226 ± 0.00	0.446 ± 0.00	0.631 ± 0.00	**0.865 ± 0.00**	0.211 ± 0.00	0.576 ± 0.00
2	0.573 ± 0.01	0.750 ± 0.00	0.541 ± 0.00	0.768 ± 0.01	**0.636 ± 0.00**	0.861 ± 0.00	0.550 ± 0.00	0.904 ± 0.00
3	**0.589 ± 0.00**	0.760 ± 0.01	**0.806 ± 0.00**	**0.889 ± 0.00**	0.604 ± 0.00	0.836 ± 0.00	**0.987 ± 0.00**	**0.999 ± 0.00**
Avg.	0.581 ± 0.00	0.758 ± 0.00	0.524 ± 0.00	0.701 ± 0.00	0.624 ± 0.00	0.854 ± 0.00	0.582 ± 0.00	0.826 ± 0.00

(b) FedProx, GRU-Spatial, 6,000 users and 4 clients.

Table 10: Impact of data distribution on accuracy results (with 95% confidence intervals) in scenarios of Table 6.