Differentially Private Multivariate Time Series Forecasting of Aggregated Human Mobility With Deep Learning: Input or Gradient Perturbation?

Héber H. Arcolezi[1,2] (iD) · Jean-François Couchot[2] (iD) · Denis Renaud[3] (iD) · Bechara Al Bouna[4] (iD) · Xiaokui Xiao[5] (iD)

Abstract This paper investigates the problem of forecasting multivariate aggregated human mobility while preserving the privacy of the individuals concerned. Differential privacy, a state-of-the-art formal notion, has been used as the privacy guarantee in two different and independent steps when training deep learning models. On one hand, we considered *gradient perturbation*, which uses the differentially private stochastic gradient descent algorithm to guarantee the privacy of each time series sample in the learning stage. On the other hand, we considered *input perturbation*, which adds differential privacy guarantees in each sample of the series before applying any learning. We compared four state-of-the-art recurrent neural networks: Long Short-Term Memory, Gated Recurrent Unit, and their Bidirectional architectures, i.e., Bidirectional-LSTM and Bidirectional-GRU. Extensive experiments were conducted with a real-world multivariate mobility dataset, which we published openly along with this paper. As shown in the results, differentially private deep learning models trained under gradient or input perturbation achieve nearly the same performance as non-private deep learning models, with loss in performance varying between 0.57% to 2.8%. The contribution of this paper is significant for those involved in urban planning and decision-making, providing a solution to the human mobility multivariate forecast problem through differentially private deep learning models.

1 Introduction

Efficiently planning a road network, choosing the optimal location for a hospital, for example, are all decisions based on a precise understanding of human mobility. Mobile phone data such as call detail records (CDRs) have proven to be one of the most promising ways to analyze human mobility on a large scale due to the high penetration rates of cell phones [35,14,13]. CDR is a type of metadata that describes users' activities in a cellular network (e.g., phone calls, SMS) with information such as the duration of communication, the antennas that handled the service (coarse level location), and so on. For this reason, CDRs are commonly used by mobile network operators (MNOs) to enhance their services and for billing and legal purposes [39,13].

Because both temporal and spatial information is available in CDRs, these data have become one of the most important data sources for research on human mobility [14,30,35,8,20]. Indeed, human mobility analysis can benefit individuals and society enabling local authorities to improve urban planning, enhance the transportation system, and assist in decision-making to respond to critical situations (e.g., natural disasters [20, 28]). Within a recent context, due to the ongoing Coronavirus Disease 2019 (COVID-19) pandemic [56], on 8 April 2020, the European Commission asked MNOs in the European region to share anonymized and aggregated mobility data to help to fight the outbreak [54,

✉ Héber H. Arcolezi
    heber.hwang-arcolezi@inria.fr

[1]Inria and École Polytechnique (IPP), Palaiseau, France
[2]Femto-ST Institute, Univ. Bourg. Franche-Comté, UBFC, CNRS, Belfort, France
[3]Orange Applications For Business, Orange Labs., Belfort, France
[4]TICKET Lab., Antonine University, Hadat-Baabda, Lebanon
[5]School of Computing, National University of Singapore, Singapore

24], which has also been done in other parts of the world as described in [8]. This vision is also shared by, e.g., Buckee et al. [14] and Oliver et al. [39], which highlight the importance of aggregate mobility data and mobile phone data like CDRs for fighting the COVID-19 outbreak.

For instance, on analyzing the dataset at our disposal (further explained in Subsection 2.1), Fig. 1 illustrates aggregated mobility analytics in Paris, France, for two 14-days periods: from the beginning of 2020-04-21 to the end of 2020-05-03 and from the beginning of 2020-09-23 to the end of 2020-10-06. The plot on the left-side corresponds to mobility analytics during the first national lockdown period in France [1], and the plot on the right-side corresponds to a period with no lockdown measures. As one can notice, there is a clear difference between the first period of analysis (low mobility activity) and the second one (high mobility activity). This type of mobility analysis provides important insights on mobility patterns for public authorities and policymakers, for example [54,8].

However, on analyzing mobility data, some studies have shown that humans follow particular patterns with a high identifiability [36,38] and, hence, *users' location privacy is a major concern* [36,38,11,35,7,5]. Indeed, even though in CDRs the location information is at a coarse level (antennas that handled the service), collecting many imprecise locations can still lead to privacy leaks, such as the home or work addresses. Also, this is a scenario in which users cannot sanitize their data locally since CDRs are automatically generated on MNOs' servers through the use of a service (e.g., making/receiving phone calls). To tackle these issues, the General Data Protection Regulation (GDPR) [3] as well as some data protection authorities, such as the Commission Nationale de l'Informatique et des Libertés (CNIL) [2], in France, require that MNOs anonymize "on-the-fly" CDRs used for purposes other than billing. More precisely, if CDRs are used for mobility analytics, these data must be processed within a required time interval (e.g., 15 minutes) if and only if there is a sufficient number of users present for reaching a specific level of anonymity (i.e., "hide in the crowd").

This way, MNOs tend to publish aggregated mobility data [8,57,54,53,40], e.g., the number of users by coarse location at a given timestamp, which, in other words, represents a *multivariate time series dataset* that can be used for predictive mobility [30]. Nevertheless, as recent studies have shown, even aggregated mobility data can be subject to membership inference attacks [42,43] and users' trajectory recovery attack [53, 57], thus requiring proper sanitization. To tackle privacy concerns in data releases, research communities have proposed different methods to preserve privacy, with differential privacy (DP) [21,22] standing out as a formal definition that allows quantifying the privacy-utility trade-off. Differential privacy has also been at the core of many privacy-preserving machine learning (ML) and deep learning (DL) models [48,4,60,17,31, 33] since predictive models are also subject to privacy attacks [16,58,50,15,49].

With these elements in mind, this paper contributes with a comparative analysis between adding DP guarantees into two different steps of training DL models to forecasting multivariate aggregated human mobility data. On the one hand, we consider using *gradient perturbation*, which can be achieved by training DL models over original time-series data with the differentially private version of the stochastic gradient descent algorithm (DP-SGD) [4]. On the other hand, we consider using *input data perturbation*, i.e., training DL models with differentially private time series data. Notice that, while aggregated time-series data provides some anonymity-based protection, with the latter input perturbation setting, DP also adds a layer of protection against, e.g., data breaches [32], membership inference attacks [42,43], and users' trajectory recovery attacks [53,57].

It is worth mentioning that human mobility forecast information is of great importance for public and/or private organizations to identify strategies to propose better decision-making solutions for society [55,35,8, 39,20,28,14,54]. Therefore, in this paper, extensive experiments were carried out with a real-world mobility dataset collected by Orange [40] on analyzing CDRs in 6 coarse regions in Paris, France, which we publish openly as an open mobility dataset. More precisely, this paper benchmarks four state-of-the-art DL models with this dataset, i.e., recurrent neural networks (RNNs): Long Short-Term Memory (LSTM) [27], which is capable of learning long-term dependencies while overcoming the vanishing gradient problem of standard RNNs; Gated Recurrent Unit (GRU) [19], which is similar to LSTM but with a simpler architecture; and their Bidirectional [46] architectures, i.e., BiLSTM and BiGRU. Moreover, we also took into consideration users' privacy, adding DP guarantees into the predictive models and evaluating their utility loss in comparison with non-private DL models.

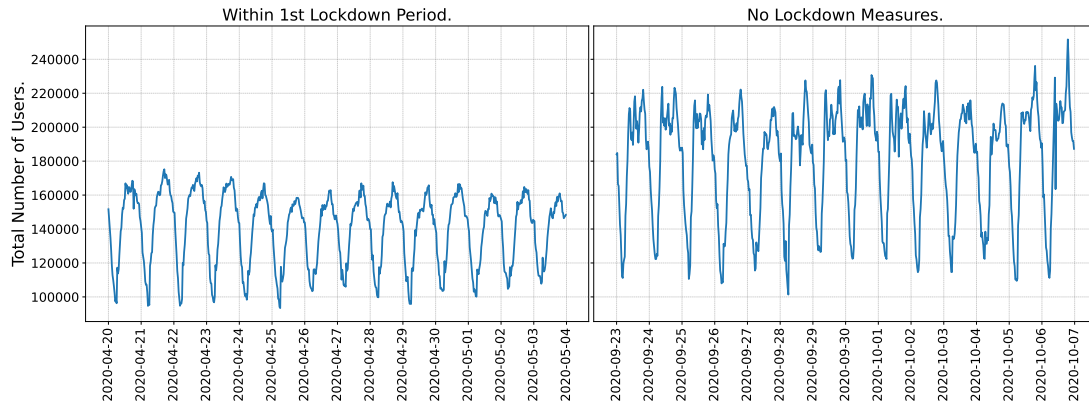**To summarize, this paper makes the following contributions:**

Fig. 1: Aggregated human mobility analytics in Paris during the COVID-19 pandemic: two weeks within the first lockdown period in France (left-side plot) and during two weeks with no lockdown measures (right-side plot).

- Publish the real-world, CDRs-based, and multivariate (i.e., 6 coarse regions) aggregated mobility dataset openly in a Github repository[1].
- Benchmark four state-of-the-art RNNs (LSTM, GRU, BiLSTM, and BiGRU) with this dataset for *one-step-ahead* multivariate forecasting.
- Provide the first comparative evaluation on the impact of differential privacy guarantees when training DL models in both input and gradient perturbation settings, for multivariate time series forecasting.

Therefore, we intend that from this study, other classical time series forecasting, ML, and privacy-preserving ML techniques can be tested and compared. **Outline.** The remainder of this paper is organized as follows. In Section 2, we describe the material and methods used in this work, i.e., the mobility data and the problem statement; the DL methods, and the privacy guarantee, namely, differential privacy for both input and gradient perturbation settings. In Section 3, we present the experimental setup, our results with non-private DL models, and our results with differentially private DL models. In Section 4, we discuss our work with its limitations and future directions. Lastly, in Section 5, we present the concluding remarks.

2 Material and Methods

In this section, we first describe the mobility dataset and the problem we intend to solve (Subsection 2.1). Next, we briefly describe the DL methods we consider in our experiments (Subsection 2.2). Lastly, we recall the privacy notion that we are considering, i.e., differ-

ential privacy, in both input and gradient perturbation settings (Subsection 2.3).

2.1 Mobility Dataset and Problem Statement

The dataset at our disposal was provided by an MNO in France [40], which contains anonymized and aggregated human mobility data resulted from analyzing CDRs "on-the-fly", following recommendations from both GDPR and CNIL. This dataset comprises information for two periods: from 2020-04-20 to 2020-05-03 and from 2020-08-24 to 2020-11-04, with time granularity of 30 minutes (min) and spatial granularity of 6 coarse regions in Paris, France.

More formally, this is a multivariate time series dataset $X_{(t_1, t_\tau)}$ with aggregate number of people per region and corresponding time period $t \in [1, \tau]$. That is, $X_{(t_1, t_\tau)} = [\langle t_1, \mathbf{x}_1 \rangle, \langle t_2, \mathbf{x}_2 \rangle, ..., \langle t_\tau, \mathbf{x}_\tau \rangle]$, in which $\mathbf{x}_t$ is a vector with each position representing the number of users per region at time $t \in [1, \tau]$. In this paper, we aim at forecasting the future number of people at the next 30-min interval in each of the 6 regions. Thus, given $X_{(t_1, t_\tau)}$, the goal is to forecast $X_{(t_{\tau+1})}$, i.e., *one-step-ahead* forecasting, which is unknown at time τ.

For the rest of this paper, we only utilize the second period of this dataset (i.e., from 2020-08-24 to 2020-11-04), which has aggregated mobility data for 72 days. For each week, coarse region, and 30-min interval, we used the interquartile range technique[2] to detect outliers and missing data. These values were completed with the average value for that respective week, coarse region, and 30-min interval. Table 1 presents descriptive statistics about this processed dataset with the following measures per region (labeled as R1 - R6): min, max, mean, standard deviation (std), and median.

Table 1: Descriptive statistics for the multivariate time series dataset on the number of users per coarse region.

Statistic	R1	R2	R3	R4	R5	R6
Min	56,937	1,996	1,429	255	252	347
Max	165,405	21,980	28,990	25,184	7,961	27,637
Mean	116,777	14,307	16,274	11,758	4,166	11,559
Std	17,947	2,803	3,915	3,682	1,450	5,136
Median	121,488	14,808	16,661	12,134	4,495	12,542

2.2 Deep Learning Methods

To predict the number of users in each coarse region in a multivariate time series forecasting framework, we compared the performance of four state-of-the-art RNNs: LSTM [27], GRU [19], and their Bidirectional [46] architectures, i.e., BiLSTM and BiGRU. Indeed, RNNs is a specialized class of neural networks used to process sequential data (e.g., time-series data). RNNs have at least one feedback connection that provides the ability to use contextual information when mapping between input and output sequences [26]. The LSTM, GRU, and Bidirectional RNN methods are briefly described in the following subsections.

2.2.1 Long Short-Term Memory

Long Short-Term Memory [27] is a type of RNN that overcomes the vanishing gradient problem of standard RNNs. The memory cell of LSTM divides its states in long-term state $c_{(t)}$ and short-term state $h_{(t)}$. The learning process is controlled by three gates: input $i_{(t)}$, forget $f_{(t)}$ and output $o_{(t)}$ gates, which give LSTM the ability to learn which data in a sequence is important to keep or to discard. More precisely, both input $x_{(t)}$ and the previous short-term state $h_{(t-1)}$ are fed to four different and fully connected layers. Then, the first layer computes the internal hidden state $g_{(t)}$, using $x_{(t)}$ and $h_{(t-1)}$, and partially store $g_{(t)}$ in the long-term state. The remaining tree layers are the nonlinear gating units. The forget gate $f_{(t)}$ controls the past information which must be vanished or kept. The input gate $i_{(t)}$ controls the new information which is to be added to the long-term state. Lastly, the output gate $o_{(t)}$ controls which information could be utilized for the output of the memory cell $y_{(t)}$. The mathematical formulation is as follows [27]:

$$i_{(t)} = \sigma(W_{xi}x_{(t)} + W_{hi}h_{(t-1)} + b_i),$$
$$f_{(t)} = \sigma(W_{xf}x_{(t)} + W_{hf}h_{(t-1)} + b_f),$$
$$o_{(t)} = \sigma(W_{xo}x_{(t)} + W_{ho}h_{(t-1)} + b_o),$$
$$g_{(t)} = \tanh(W_{xg}x_{(t)} + W_{hg}h_{(t-1)} + b_g),$$
$$c_{(t)} = f_{(t)} \otimes c_{(t-1)} + i_{(t)} \otimes g_{(t)},$$
$$y_{(t)} = h_{(t)} = o_{(t)} \otimes \tanh(c_{(t)}),$$

in which $\otimes$ means an element-wise multiplication, σ is the sigmoid function, W_* are weight matrices, and b_* are the vectors of bias term.

2.2.2 Gated Recurrent Unit

Gated Recurrent Unit [19] is also a type of RNN, which works using the same principle as LSTM. GRU utilizes two gates: update $z_{(t)}$ and reset $r_{(t)}$, which decide what information should be passed to the output. More specifically, the reset gate $r_{(t)}$ controls how to combine the new input with the previous memory. The update gate $z_{(t)}$ controls how much of the last memory to keep. The mathematical formulation is as follows [19]:

$$z_{(t)} = \sigma(W_z x_{(t)} + U_z h_{(t-1)}),$$
$$r_{(t)} = \sigma(W_r x_{(t)} + U_r h_{(t-1)}),$$
$$c_{(t)} = \tanh(W x_{(t)} + U(r_{(t)} \otimes h_{(t-1)})),$$
$$h_{(t)} = (1 - z_{(t)})h_{(t-1)} + (z_{(t)}c_{(t)}).$$

2.2.3 Bidirectional RNN

Bidirectional RNN (BiRNN) [46] is a combination of two RNNs: one RNN moves forward while the other moves backward. That is, BiRNN connects two hidden layers of opposite directions to the same output. The RNN cells in a BiRNN can either be standard RNNs, LSTMs, GRUs, and so on. This Bidirectional architecture allows the networks to have both backward and forward information about the sequence at every time step.

2.3 Differential Privacy

In recent years, differential privacy [21] has been increasingly accepted as the current standard for data privacy with several large-scale implementations in the real-world (e.g. [45,6]). One key reason is that DP addresses the paradox of learning about a population while learning nothing about single individuals [22]. More specifically, the idea is that removing (or adding) a single row from the database should not affect *much* the statistical results. A formal definition of DP is given in the following [22]:

Definition 1 ((ϵ, δ)-Differential Privacy [22]) Given $\epsilon > 0$ and $0 \leq \delta < 1$, a randomized algorithm $\mathcal{A} : \mathcal{D} \rightarrow R$ is said to provide (ϵ, δ)-differential-privacy ((ϵ, δ)-DP) if, for all neighbouring datasets $D_1, D_2 \in \mathcal{D}$ that differ on the data of one user, and for all sets R of outputs:

$$\Pr[\mathcal{A}(D_1) \in R] \leq e^\epsilon \Pr[\mathcal{A}(D_2) \in R] + \delta. \quad (1)$$

The additive δ is interpreted as a probability of failure. Normally, a common choice for δ is to set it significantly smaller than $1/n$ where n is the number of users in the database [22]. Throughout this paper, if $\delta = 0$, we will just say that $\mathcal{A}$ is ϵ-DP.

2.3.1 Properties of Differential Privacy

Differential privacy possesses several important properties, highlighting its strength in comparison with other privacy models. For instance, with DP, there is no need to define the *background knowledge* that attackers might have, which is equivalent to assuming an attacker with *unlimited resources*. In addition, DP is immune to *post-processing*, which means it is not possible to make an ϵ-DP mechanism less differentially private by evaluating any function f of the response of the mechanism, given that there is no additional information about the database.

Proposition 1 (Post-Processing of DP [22]) *If $\mathcal{A} : \mathcal{D} \rightarrow R$ is ϵ-DP, then $f(\mathcal{A})$ is also ϵ-DP for any function f.*

Furthermore, DP also *composes* well, which is one of the most powerful features of this privacy model. For instance, accounting for the *overall* privacy loss consumed in a pipeline of several DP algorithms applied to the same database is feasible due to composition. We recall the sequential composition that will be used in this paper in the following.

Proposition 2 (Sequential Composition [22]) *Let $\mathcal{A}_1$ be an ϵ_1-DP mechanism and $\mathcal{A}_2$ be an ϵ_2-DP mechanism. Then, the mechanism $\mathcal{A}_{1,2}(\mathcal{D}) = (\mathcal{A}_1(\mathcal{D}), \mathcal{A}_2(\mathcal{D}))$ is ($\epsilon_1 + \epsilon_2$)-DP.*

2.3.2 Differentially Private Gaussian Mechanism

Any mechanism that respects Definition 1 can be considered differentially private. Two widely used DP mechanisms for numeric queries (i.e., functions $f : \mathcal{D} \rightarrow \mathbb{R}$) are the Laplace mechanism [21] and the Gaussian mechanism [22]. One important parameter that determines how accurately we can answer the queries is their *sensitivity*. We recall the definition of ℓ_2-sensitivity and

the Gaussian mechanism that will be used in this paper in the following.

Definition 2 (ℓ_2-sensitivity [22]) The ℓ_2-sensitivity of a function $f : \mathcal{D} \rightarrow \mathbb{R}$, for all neighbouring datasets $D_1, D_2 \in \mathcal{D}$ that differ on the data of one user, is:

$$\Delta_2(f) = max \, ||f(D_1) - f(D_2)||_2.$$

Definition 3 (Gaussian mechanism [22]) For a query function $f : D \rightarrow \mathbb{R}$ over a dataset $D \in \mathcal{D}$ and for $\sigma = \frac{\Delta_2}{\epsilon}\sqrt{2 \ln{(1.25/\delta)}}$, the Gaussian mechanism is defined as:

$$\mathcal{A}_G(D, f(.), \epsilon, \delta) = f(D) + \mathcal{N}\left(0, \sigma^2\right).$$

in which $\mathcal{N}\left(0, \sigma^2\right)$ is the normal distribution centered at 0 with variance σ^2. For $\epsilon \in (0, 1)$, the Gaussian mechanism provides (ϵ, δ)-DP [22].

2.3.3 Differentially Private Deep Learning

In this paper, on the one hand, we consider that noise is added to each sample in the time series data (input perturbation). Once the data is differentially private, following Proposition 1, any DL or pre-processing methods can be applied to the data. On the other hand, we consider the case where noise is added in the learning stage (gradient perturbation). In this case, the raw time-series data is used as input to a DL method trained with the differentially private stochastic gradient descent algorithm. These two settings are briefly described in the following:

Input data perturbation. Input perturbation (or data perturbation) consists to the fact that *DP is added to each data sample* $\mathbf{x}_i \in \mathcal{D}$. For example, let $\mathbf{x}$ be a real-valued vector, then a differentially private version of it using the Gaussian mechanism (cf. Subsection 2.3.2) is: $\hat{\mathbf{x}} = \mathbf{x} + \mathcal{N}\left(0, \sigma^2\right)$. On the one hand, input perturbation is the easiest method to apply and it is independent of any ML and post-processing techniques. On the other hand, the perturbation of each sample in the dataset may have a negative impact on the utility of the trained model. In this paper, we will use the Gaussian mechanism [22] in Definition 3 to sanitize each release of multivariate mobility data.

Gradient perturbation. Another solution to guarantee DP to the trained model is perturbing intermediate values in iterative algorithms. In this case, the authors in [4] proposed a differentially private version of the stochastic gradient descent algorithm (i.e., DP-SGD). Indeed, DL models trained with DP-SGD [4] provide provable DP guarantees for their input data. Two new parameters are added to the standard stochastic gradient descent algorithm, namely, *clip* and *noise multiplier*. The former is used to bound how much each

training point can impact the model's parameters and the latter is used to add controlled Gaussian noise to the clipped gradients in order to ensure DP guarantee to each data sample in the training dataset. In this paper, we use the DP-SGD implementation from the Tensorflow Privacy (TFP) library [33] to implement our DL models.

3 Experimental Results

We divide this section in the following way. First, we describe general settings for our experiments (Subsection 3.1). Next, we present the development and evaluation of non-private DL models (Subsection 3.2). Lastly, we present the development of differentially private DL models, which include both gradient and input perturbation settings (Subsection 3.3).

3.1 General Setup of Experiments

Environment. All algorithms were implemented in Python 3.8.8 with Keras and TFP [33] libraries.

Temporal features. We added the time of the day and the time of the week as cyclical features to help models recognizing low and high peak values of human mobility patterns.

Training and testing sets. We split the dataset analyzed in Table 1 from Subsection 2.1 into exclusively divided learning (first 65 days, i.e., $n_l = 3120$ intervals of 30-min) and testing (last 7 days, i.e., $n_t = 336$ intervals of 30-min) sets. Fig. 2 exemplifies the data separation into train and test sets for region R1.

Forecasting methodology. We used 6 prior time steps (i.e., lag values), which showed autocorrelation higher than 0.5 to predict a single step ahead in the future (i.e., short forecasting horizon). More specifically, the forecasting models will take into account the number of people in each coarse region from 3 hours to make predictions *one-step-ahead* for each coarse region in the next 30-min interval. And in the end, we compute the performance metrics.

Performance metrics. All models were evaluated with standard time-series metrics, namely, root mean square error (RMSE) calculated as $RMSE = \frac{1}{n_t}\sqrt{\sum_{t=1}^{n_t}(y_t - \hat{y}_t)^2}$ and mean absolute error (MAE) calculated as $MAE = \frac{1}{n_t}\sum_{t=1}^{n_t}|y_t - \hat{y}_t|$; in which y_t is the real output, $\hat{y}_t$ is the predicted output, and n_t is the total number of samples in the *test set*, for $t \in [1, n_t]$. RMSE was the primary metric to select the final DL models. As a multi-output scenario (i.e., 6 coarse regions), we present the metrics per coarse region as well

as its averaged values. In all experiments, due to randomness, we report the results of the best model over 10 runs.

3.2 Non-Private DL Forecasting Models

Baseline model. We compared the performance of the four DL methods (i.e., LSTM, GRU, BiLSTM, and BiGRU) with a naive forecasting technique a.k.a. "*persistence model*", which for each region, it returns the current number of people at time t as the forecasted value, i.e., $\mathbf{x}_{t+1} = \mathbf{x}_t$. Notice that this is a quite accurate baseline since, in general, the number of people per coarse region varies slowly by 30-min (i.e., walking people may take more time to move from one coarse region to another).

Model selection. To select the best hyperparameters per DL method, we used Bayesian optimization [12] with 100 iterations to minimize $loss = RMSE_{avg} + RMSE_{std}$; the subscripts *avg* and *std* indicates the averaged and standard deviation values of the RMSE metric considering the 6 coarse regions. For each method, we only used a single hidden layer followed by a dense layer (output), since RNNs generally perform well with a low number of hidden layers [26]. So, we searched the following hyperparameters: number of neurons (h_1), batch size (bs), and learning rate (η). All models used "relu" (rectified linear unit) as an activation function, which resulted in better performance than the default "tanh" activation in prior tests. Lastly, models were trained using the adam (adaptive moment estimation) optimizer during 100 epochs by minimizing the MAE loss function. Table 2 exhibits the hyperparameters' search space and the final value used per DL method.

Results and analysis. Table 3 presents the performance of the developed DL models in comparison with the Baseline model based on RMSE and MAE metrics per region and the resulting mean. Notice that the metrics are in the real scale according to the number of users per region (cf. Table 1). That said, although R1 presents higher metric values, it does not necessarily mean worse results. One solution could be normalizing the data. Besides, Fig. 3 illustrates for each region forecasting results for the last day of our testing set, which includes the real number of people and the predicted ones by each RNN: LSTM, GRU, BiLSTM, and BiGRU.

As one can notice, all DL models consistently outperform the Baseline model. On average, the BiGRU model outperformed all other forecasting methods, with results highlighted **in bold** in Table 3. Indeed, for each region, the BiGRU consistently and considerably outperformed the Baseline model, showing the worthiness
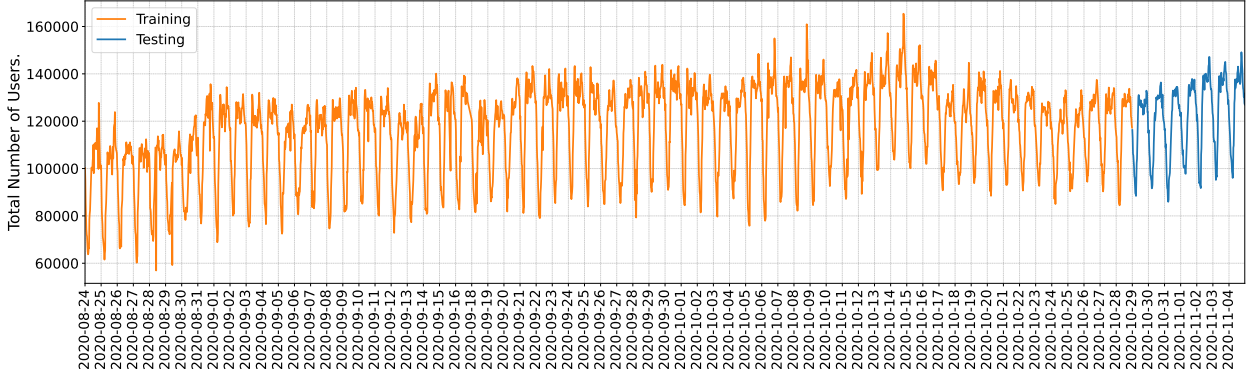
Fig. 2: Example of data separation into training and testing sets for region R1.

Table 2: Search space for hyperparameters and the best configuration obtained by each DL method.

Hyperparameter's range	Step	LSTM	BiLSTM	GRU	BiGRU
h_1: [25 – 500]	25	225	500	75	175
bs: [5 – 40]	5	10	10	5	5
η: [1e-5 – 3e-3]	–	0.002233	0.002303	0.001725	0.000289

Table 3: Performance of the Baseline model and non-private DL models based on RMSE and MAE metrics per region and the resulting mean values.

Model	Metric	R1	R2	R3	R4	R5	R6	Mean
Baseline	RMSE	3461.6	1131.8	1517.9	986.5	561.3	1362.3	1503.6
	MAE	2597.5	839.4	1105.8	744.1	434.3	921.5	1107.1
LSTM	RMSE	2667.2	1007.3	1291.6	887.2	536.3	1135.6	1254.2
	MAE	2053.8	758.1	969.8	662.6	432.3	786.0	943.8
BiLSTM	RMSE	2572.7	1033.3	1276.4	872.7	528.1	1166.7	1241.6
	MAE	1954.7	781.5	965.5	660.8	419.4	808.2	931.7
GRU	RMSE	2539.1	973.0	1296.0	953.5	499.9	1185.1	1241.1
	MAE	1949.7	722.8	939.6	740.2	396.4	829.1	929.6
BiGRU	RMSE	2560.3	968.3	1282.6	832.1	478.9	1163.7	**1214.3**
	MAE	1957.2	717.0	955.3	623.0	382.7	807.5	**907.1**

of developing DL models for this multivariate forecasting task. Similar scores were achieved by the GRU and BiLSTM models with an average RMSE around 1241. The least performing DL method in our dataset was the LSTM model. Extending the architectures, hyperparameters range, lag values (i.e., test with less or more input time steps), dropout layers, for example, could probably improve our models and change the resulting best technique. However, we will focus our attention on a comparative analysis of differentially private DL methods in the next subsection and, thus, these possible extensions are left as future work.

3.3 Differentially Private DL Forecasting Models

Methods evaluated. We consider two privacy-preserving ML settings presented in Subsection 2.3, namely, input perturbation (IP) and gradient perturbation (GP). Thus, we selected only the DL method that performed the best with original data, i.e., BiGRU (cf. Table 3). We will use BiGRU[IP] and BiGRU[GP] to indicate a BiGRU trained under input and gradient perturbation, respectively.

For the model selection stage, we first start with BiGRU[GP] since it allows defining a range of ϵ, which is dependent on several hyperparameters of DP-SGD. For a fair comparison between both settings, we utilize the given range of ϵ to develop BiGRU[IP] models too. Notice, however, that in both scenarios, (ϵ, δ)-DP can be ensured to each time series data sample. On the other hand, this also means that the same user may have contributed to all $n_l = 3120$ training samples and, thus, in the worst case, the sequential composition in Proposition 2 applies. With these elements in mind, we considered high privacy regimes ($\epsilon \ll 1$) such that the maximum $\tilde{\epsilon} = \sum_{i=1}^{n_l} \epsilon_i$ is compatible with real-world DP deployed systems [45, Table 2]. **This way, ϵ corresponds to the lower bound (the user appears**
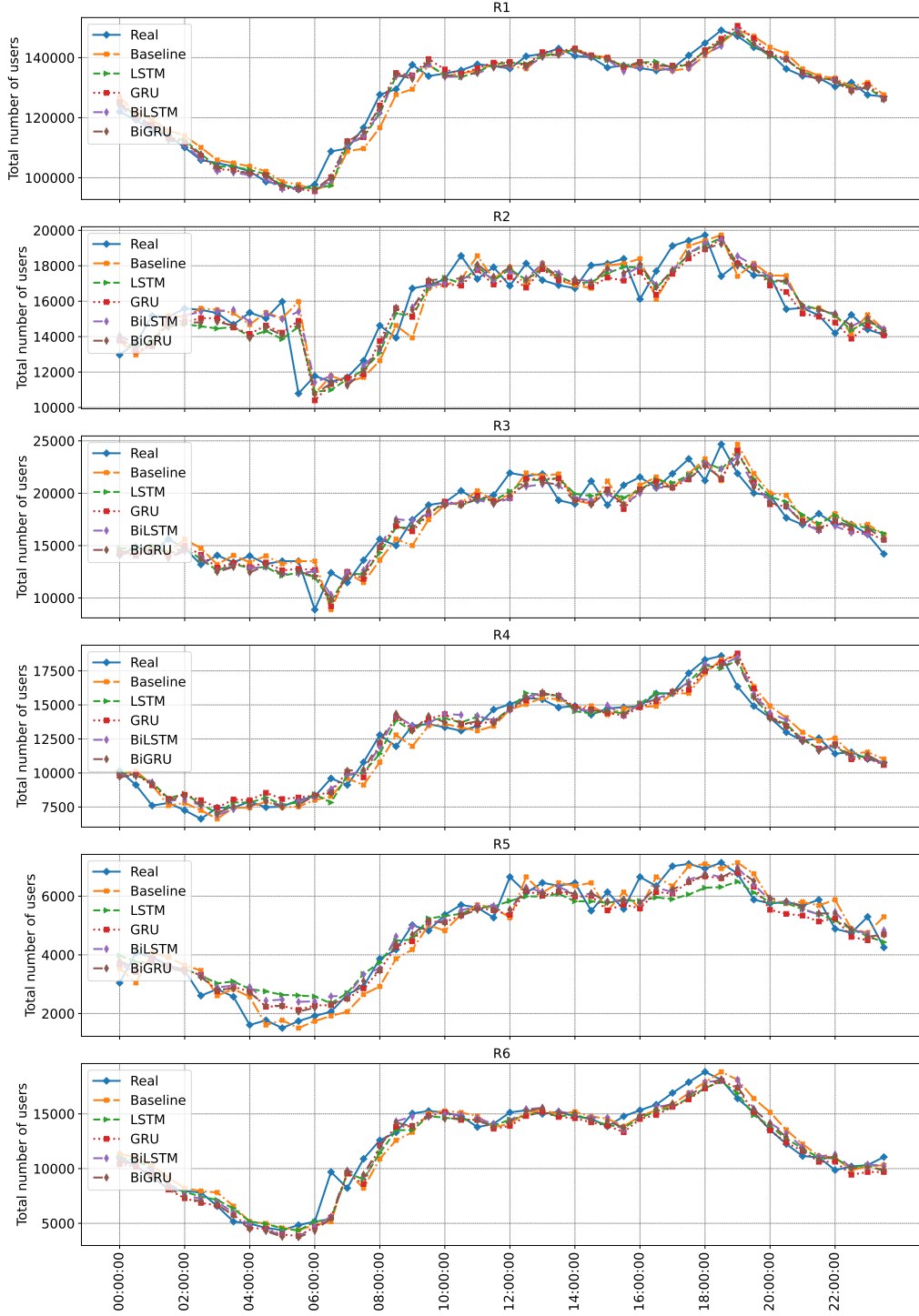
Fig. 3: Multivariate time series forecast for the last day of the test set for the number of users per coarse region (R1 – R6) by the following models: Baseline, LSTM, GRU, BiLSTM, and BiGRU.

in a single data point), and $\check{\epsilon}$ represents the upper bound (the user appears in all data points).

BiGRU[GP] model selection. In addition to standard hyperparameters h_1, bs, and η (cf. Subsection 3.2), we also included the TFP hyperparameters in the

Bayesian optimization with 100 iterations to minimize $loss = (RMSE_{avg} + RMSE_{std}) \times e^{\epsilon}$; the multiplicative factor e^{ϵ} is a penalization on high values of ϵ, which varies depending on the hyperparameters used per iteration. More specifically, given

the number of training samples $n_l = 3120$, we fix the following hyperparameters: the number of epochs equal 100, $num\_microbatches = 5$, $noise\_multiplier$ equal $\{35, 70, 140, 500\}$, respectively, and $\delta = 10^{-7}$, which respects $\sum_{i=1}^{n_l} \delta_i < 1/n_l$ [22]. This way, we varied h_1, bs, η, and $l2\_norm\_clip$ according to Table 4, which exhibits the hyperparameters' search space, the final value used per BiGRU[GP] model, and the resulting privacy guarantee ϵ calculated with the `compute_dp_sgd_privacy` function [33], and the overall $\check{\epsilon} = \sum_{i=1}^{n_l} \epsilon_i$. Lastly, all BiGRU[GP] models also used "relu" as an activation function and were trained using the differentially private adam optimizer by minimizing the MAE loss function.

BiGRU[IP] model selection. We fix $\delta = 10^{-7}$ and we apply the Gaussian mechanism [22], by varying ϵ according to Table 4, **to the whole time series data, as it would be done if such system had been deployed in real life. The metrics, however, are computed in comparison with original raw time series data.** Because input perturbation allows using any post-processing techniques, we used the same model selection methodology as for non-private BiGRU models to select the best hyperparameters for BiGRU[IP] models. The resulting values per $\epsilon = [0.0650, 0.0399, 0.0357, 0.0317]$, respectively, are: BiGRU[IP]$_1$: $\{h_1 = 200, bs = 5, \eta = 0.001993\}$, BiGRU[IP]$_2$: $\{h_1 = 275, bs = 5, \eta = 0.001182\}$, BiGRU[IP]$_3$: $\{h_1 = 200, bs = 10, \eta = 0.001333\}$, and BiGRU[IP]$_4$: $\{h_1 = 200, bs = 10, \eta = 0.000842\}$.

Privacy-preserving results and analysis. Table 5 presents the performance of differentially private Bi-GRU models trained under input and gradient perturbation regarding the RMSE and MAE metrics per region and the resulting mean values. We also included in Table 5 the utility loss of differentially private BiGRU models in comparison with non-private ones, for both RMSE and MAE averaged metrics $\mathscr{E}$, calculated as:

$$\mathscr{U} = \frac{\mathscr{E}_{DP} - \mathscr{E}_{NP}}{\mathscr{E}_{NP}}, \tag{2}$$

in which $\mathscr{E}_{NP}$ is the result of Non-Private BiGRU (cf. averaged metric values **in bold** from Table 3) and $\mathscr{E}_{DP}$ refers to the results of either BiGRU[GP] or BiGRU[IP] models. Indeed, Eq. (2) will be positive unless the differentially private model outperforms the non-private one (which is not the case in our results).

We remarked in our experiments that since there is a sufficient number of users per time series sample (cf. Table 1), it was still possible to make accurate forecasts in both privacy-preserving ML settings with the experimented range of (ϵ, δ)-DP. Indeed, one can notice

that all differentially private BiGRU models achieved averaged RMSE lower than 1250, in which the worst result achieved by BiGRU[IP]$_4$ is just 2.8% less precise than the non-private BiGRU model. What is more, in both gradient and input perturbation settings, differentially private BiGRU models achieved smaller error metrics than non-private LSTM, BiLSTM, and GRU models (cf. Table 3). For instance, both BiGRU[GP]$_2$ and BiGRU[IP]$_2$ reached similar scores in comparison with the non-private BiGRU model, with a loss of performance of about 0.57% and 0.62%, respectively. These results are highlighted in <u>underlined</u> font, which represents our best results in terms of utility, with differentially private BiGRU models.

Interestingly, the accuracy (measured with the RMSE metric) of differentially private BiGRU models did not necessarily decrease according to more strict ϵ, i.e., lower values. One can note that results with ϵ_2 and ϵ_3 were more accurate than with ϵ_1. This way, in terms of a good *privacy-utility trade-off*, both BiGRU[GP]$_3$ (0.92% less accurate) and BiGRU[IP]$_3$ (1.53% less accurate) presented good metrics scores while satisfying a low value of ϵ, with results highlighted **in bold**. Indeed, in the worst-case scenario, a user that was present in each data point would have leaked $\check{\epsilon}_3 = 111.384$ at the end of 65 days (i.e., $\epsilon \sim 1.7$ per day), which follows real-world DP systems deployed by industry nowadays [45, Section 8.4].

Lastly, Fig. 4 illustrates for each region forecasting results for the last day of our testing set, which includes the real number of people and the predicted ones by the following models: Baseline, non-private BiGRU, BiGRU[GP]$_3$, and BiGRU[IP]$_3$. As one can notice, similar forecasting results were achieved by both non-private and DP-based BiGRU models, which clearly outperforms the Baseline model.

4 Discussion and Related Work

Mobile phone CDRs have been largely used to analyze human mobility in several contexts, e.g., the spread of infectious diseases [55,8], natural disasters [28,20], tourism [40], and so on. However, on analyzing mobility data, de Montjoye et al. [36] show that humans follow particular patterns, which allows predicting human mobility with high accuracy. For instance, in a CDRs dataset of 1.5 million users, the authors showed that 95% of this population can be re-identified using four approximate locations and their timestamps. Indeed, the uniqueness of mobility data has also been studied in [38], for example, in which the authors concluded that location trace has higher identifiability than a face matcher in a partial-knowledge model.

Table 4: Search space for standard and TFP hyperparameters, the best configuration per BiGRU[GP] model, the final privacy guarantee ϵ per time-series sample, and the maximum $\check{\epsilon}$ following the sequential composition in Proposition 2.

Hyperarameter's range	Step	BiGRU[GP]$_1$	BiGRU[GP]$_2$	BiGRU[GP]$_3$	BiGRU[GP]$_4$
h_1: [25 − 500]	25	500	425	275	475
bs: [5 − 40]	5	5	5	10	5
η: [1e-5 − 3e-3]	−	0.002229	0.000455	0.000291	0.001235
$l2\_norm\_clip$: {1, 1.5, 2, 2.5}	−	2.5	2	1	2.5
$noise\_multiplier$: `fixed`	−	35	70	140	500
Privacy guarantee		$\epsilon_1 = 0.0650$	$\epsilon_2 = 0.0399$	$\epsilon_3 = 0.0357$	$\epsilon_4 = 0.0317$
		$\check{\epsilon}_1 = 202.8$	$\check{\epsilon}_2 = 124.488$	$\check{\epsilon}_3 = 111.384$	$\check{\epsilon}_4 = 98.904$

Table 5: Performance of differentially private BiGRU models based on RMSE and MAE metrics per region and the resulting mean values. The last column $\mathscr{U}$ exhibits the utility loss of differentially private BiGRU models in comparison with non-private ones, for both RMSE and MAE averaged metrics expressed in %.

$\epsilon, \check{\epsilon}$ values	Model	Metric	R1	R2	R3	R4	R5	R6	Mean	$\mathscr{U}$
$\epsilon_1 = 0.0650$	BiGRU[GP]$_1$	RMSE	2561.4	1027.3	1254.7	866.7	498.5	1145.7	1225.7	0.9378
		MAE	1973.4	773.8	925.	644.1	397.9	781.5	916.0	0.9776
$\check{\epsilon}_1 = 202.8$	BiGRU[IP]$_1$	RMSE	2600.9	997.1	1304.0	852.7	483.8	1175.2	1235.6	1.7531
		MAE	1966.0	737.5	957.1	645.4	385.1	821.1	918.7	1.2753
$\epsilon_2 = 0.0399$	BiGRU[GP]$_2$	RMSE	2600.2	956.0	1268.5	841.5	515.0	1146.3	<u>1221.2</u>	<u>0.5672</u>
		MAE	1978.9	709.2	944.4	643.3	417.4	769.9	910.5	0.3713
$\check{\epsilon}_2 = 124.488$	BiGRU[IP]$_2$	RMSE	2592.2	978.4	1251.5	854.2	495.6	1158.6	<u>1221.8</u>	<u>0.6166</u>
		MAE	1986.1	737.1	910.9	653.9	393.2	813.9	915.9	0.9666
$\epsilon_3 = 0.0357$	BiGRU[GP]$_3$	RMSE	2580.5	990.0	1268.5	854.5	504.9	1154.3	**1225.5**	**0.9213**
		MAE	1938.8	753.0	942.8	659.7	406.3	773.6	912.4	0.5808
$\check{\epsilon}_3 = 111.384$	BiGRU[IP]$_3$	RMSE	2587.8	1004.7	1262.3	843.2	512.8	1186.2	**1232.9**	**1.5307**
		MAE	1963.1	755.8	957.5	636.9	414.6	811.8	923.3	1.7824
$\epsilon_4 = 0.0317$	BiGRU[GP]$_4$	RMSE	2560.8	978.3	1322.5	836.1	494.4	1195.4	1231.3	1.3990
		MAE	1956.2	715.1	989.2	633.6	392.0	821.6	917.9	1.1871
$\check{\epsilon}_4 = 98.904$	BiGRU[IP]$_4$	RMSE	2562.2	1012.2	1351.2	862.9	533.5	1168.8	1248.4	2.8072
		MAE	1955.6	756.8	1027.6	650.1	423.9	826.8	940.2	3.6454

For this reason, MNOs tend to publish aggregated mobility data [8,57,54,53,40], which provides some form of anonymity-based protection. However, as recent studies have shown, even aggregated mobility data (e.g., heatmaps) can be subject to membership inference attacks [42,43] and users' trajectory recovery attack [53,57]. More precisely, the authors in [53,57] showed that their attack reaches accuracies as high as 73% ∼ 91%. Therefore, it is vital to design privacy-preserving techniques that allow analyzing human mobility [35].

Moreover, along with collecting time-series data, extracting meaningful forecasts is also of great interest. Time series forecasting has been a key area of ML research and application across many domains, e.g., medicine [44], finance [47], electrical power [52,37], mobility [30], and so on. However, even ML models trained with raw data can also indirectly reveal sensitive information [16,50,15,49], in particular, RNNs [58]. To protect ML models against such threats, under the state-of-the-art DP guarantee [21,22], there exist some privacy-preserving ML alternatives adopted in the literature,

e.g., input [18,31,23,29,9,25], gradient [4,33,51,60,48], and objective perturbation [17].

The contribution of our research is significant for those involved in urban planning and decision-making [35], providing a solution to the human mobility multivariate forecast problem through RNNs and differentially private BiGRUs. In addition, we point out the research community to the Github page mentioned in the introduction section, in which we release the mobility dataset used in this paper for further experimentation with time series, machine learning, and privacy-preserving methods. The related literature to our work includes the generation of synthetic mobility data [41,34,10], the development of Markov models to infer travelers' activity pattern [59], and the development of privacy-preserving methods to analyze CDRs-based data [7,11,5]. Besides, the work in [30] surveys non-private deep learning applications to mobility datasets in general. Concerning differentially private deep learning, one can find the application of gradient perturbation-based DL models for load forecasting [51], an evaluation of differentially private DL models in fed-
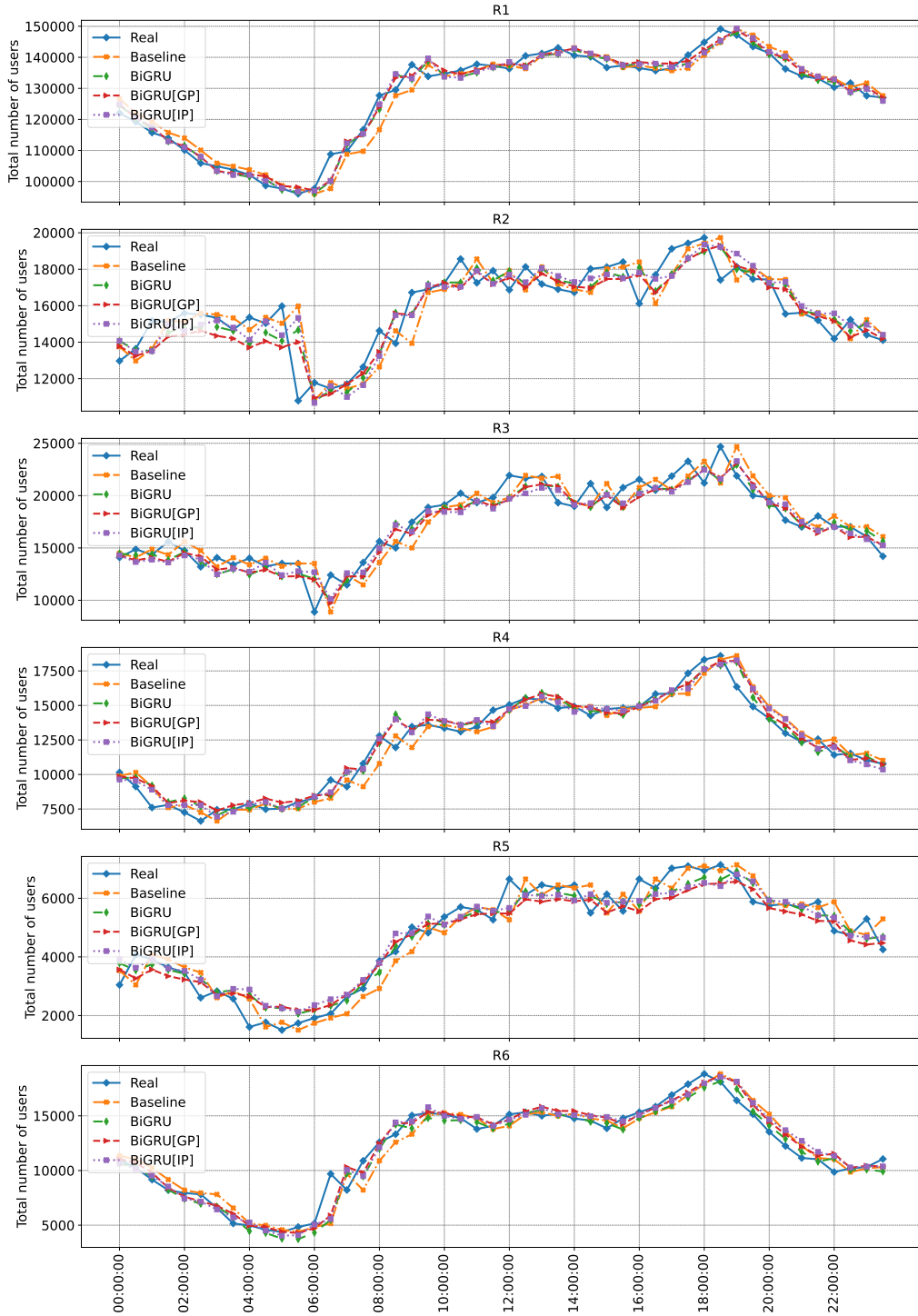
Fig. 4: Multivariate time series forecast for the last day of the test set for the number of users per coarse region (R1 – R6) by the following models: Baseline, non-private BiGRU, BiGRU[GP]$_3$, and BiGRU[IP]$_3$.

erated learning for health stream forecasting [29], the proposal of locally differentially private DL architectures [31], practical libraries for differentially private DL [33,60], and theoretical research works [4,48,18].

In this work, accurate multivariate forecasts were achieved with four non-private RNNs (i.e., LSTM, GRU, BiLSTM, and BiGRU), with BiGRU standing out among the four methods. Thus, this paper further evaluated both input and gradient perturbation

settings to forecast multivariate aggregated mobility time series data using the BiGRU neural network. Between both input and gradient perturbation settings, although not measured, BiGRU[GP] models took more time to execute than BiGRU[IP] models due to DP-SGD. In terms of accuracy, BiGRU[GP] models consistently outperformed BiGRU[IP] models for the same (ϵ, δ)-DP privacy level in our experiments. One reason for such result is because the *input data perturbation* setting adds DP guarantees to each time series point in the data, trading privacy with utility. This is indeed one fundamental problem in DP theory [22] in which local DP algorithms has low utility in comparison with centralized DP algorithms. On the other hand, as BiGRU[GP] is trained over non-DP time-series data, the mobility dataset is still subject to data leakage [32], and consequently, membership inference attacks [42,43], and users' trajectory recovery attacks [53,57], which requires strong security measures. Therefore, training ML models over differentially private multivariate time series mobility data provides the best privacy-utility trade-off. In practice, the input-perturbation setting allows applying centralized DP mechanisms in the final aggregate data (e.g., Laplace mechanism [21], Gaussian mechanism [22]), essentially refreshing the privacy budget ϵ on a regular basis, and using the published data for any purpose (cf. Proposition 1).

Finally, some limitations and prospective directions of this paper are described in the following. For differentially private BiGRU models, we only provided lower ϵ and upper $\check{\epsilon}$ bounds for the privacy guarantee of each sample in the time-series data. Using, however, advanced composition theorems [22] to account for the final privacy budget for each user was out of the scope of this paper. Indeed, CDRs are event-based [39,20], which means that data are only available when users actively make phone calls (or connect to the internet, or send SMS). This way, there may have users who make several calls (e.g., business people) and, thus, have higher values of $\check{\epsilon}$, while some groups do not, e.g., poor people. Besides, although the developed DL models outperform the Baseline model ($\mathbf{x}_{t+1} = \mathbf{x}_t$), there is plenty of room for improvements to be carried out on hyperparameters optimization, data scaling, the number of lag values, etc. For instance, some high-peak values were missed by both non-private and DP-based DL models (see Figs. 3 and 4). In addition, we fixed the number of lagged values to 6 to predict a single step-ahead in the future (i.e., the forecasting horizon), in which the former can be tuned for performance improvement and the latter can be increased for multi-step forecasting tasks. Thus, besides the aforementioned directions,

for future work, we suggest and intend to investigate a more complex DL architecture to improve the results of DL models proposed in this paper for this multivariate time series forecasting task. Lastly, investigating the data leakage through membership inference attacks [49,58] of both privacy-preserving ML settings is also a prospective and intended direction.

5 Conclusion

This paper provides the first comparative evaluation of differentially private DL models in both input and gradient perturbation settings to forecast multivariate aggregated mobility time series data. Experiments were first carried out with four non-private DL models (i.e., LSTM, GRU, BiLSTM, and BiGRU). The BiGRU model best fitted our data and, thus, it was selected for building differentially private DL models. Under gradient and input perturbation settings, i.e., BiGRU[GP] and BiGRU[IP], respectively, four values of $\epsilon \ll 1$ were evaluated. As shown in the results, differentially private BiGRU models achieve nearly the same performance as non-private BiGRU models, with loss in performance varying between 0.57% to 2.8% (for the RMSE metric). Thus, we conclude that it is still possible to have accurate multivariate forecasts in both privacy-preserving ML settings. More specifically, although the gradient perturbation setting preserved more accuracy than the input perturbation setting, input perturbation guarantees stronger privacy protection (i.e., both for the ML model and for the data itself), thus providing the best privacy-utility trade-off.

Declarations

Funding

Conflict of Interest

The authors have no conflicts of interest to declare that are relevant to the content of this article.

Availability of data and material

Data can be accessed on the following Github page (`https://github.com/hharcolezi/ldp-protocols-mobility-cdrs`).

Code availability

Codes are available on the following Github page (`https://github.com/hharcolezi/ldp-protocols-mobility-cdrs`).

Research involving Human Participants and/or Animals

This article does not contain any studies with human or animal subjects performed by any of the authors.

Informed Consent

As this article does not contain any studies with human participants or animals, the informed consent is not applicable.

References

1. Confinements liés à la pandémie de COVID-19 en france. Available online: `https://fr.wikipedia.org/wiki/Confinements_li%C3%A9s_%C3%A0_la_pand%C3%A9mie_de_Covid-19_en_France` (accessed on 11 July 2021)

2. Commission nationale de l'informatique et des libertés (CNIL) (1978). Available online: `https://www.cnil.fr/en/home` (accessed on 04 July 2021)

3. General data protection regulation (GDPR) (2018). Available online: `https://gdpr-info.eu/` (accessed on 04 July 2021)

4. Abadi, M., Chu, A., Goodfellow, I., McMahan, H.B., Mironov, I., Talwar, K., Zhang, L.: Deep learning with differential privacy. CCS '16, p. 308–318. Association for Computing Machinery, New York, NY, USA (2016). DOI 10.1145/2976749.2978318

5. Acs, G., Castelluccia, C.: A case study: Privacy preserving release of spatio-temporal density in Paris. In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '14. ACM Press (2014). DOI 10.1145/2623330.2623361

6. Aktay, A., Bavadekar, S., Cossoul, G., Davis, J., Desfontaines, D., Fabrikant, A., Gabrilovich, E., Gadepalli, K., Gipson, B., Guevara, M., et al.: Google COVID-19 community mobility reports: anonymization process description (version 1.1). arXiv preprint arXiv:2004.04145 (2020)

7. Alaggan, M., Gambs, S., Matwin, S., Tuhin, M.: Sanitization of call detail records via differentially-private bloom filters. In: Data and Applications Security and Privacy XXIX, pp. 223–230. Springer International Publishing (2015)

8. de Alarcon, P.A., Salevsky, A., Gheti-Kao, D., Rosalen, W., Duarte, M.C., Cuervo, C., Muñoz, J.J., Pascual, J.M., Schurig, M., Treß, T., Diaz, E., de la Cuesta, C., Frias-Martinez, E.: The contribution of telco data to fight the COVID-19 pandemic: Experience of telefonica throughout its footprint. Data & Policy **3**, e7 (2021). DOI 10.1017/dap.2021.6

9. Arcolezi, H.H., Cerna, S., Guyeux, C., Couchot, J.F.: Preserving geo-indistinguishability of the emergency scene to predict ambulance response time. Mathematical and Computational Applications **26**(3) (2021). DOI 10.3390/mca26030056

10. Arcolezi, H.H., Couchot, J.F., Baala, O., Contet, J.M., Al Bouna, B., Xiao, X.: Mobility modeling through mobile data: generating an optimized and open dataset respecting privacy. In: 2020 International Wireless Communications and Mobile Computing (IWCMC), pp. 1689–1694 (2020). DOI 10.1109/IWCMC48107.2020.9148138

11. Arcolezi, H.H., Couchot, J.F., Bouna, B.A., Xiao, X.: Longitudinal collection and analysis of mobile phone data with local differential privacy. In: M. Friedewald, S. Schiffner, S. Krenn (eds.) Privacy and Identity Management, pp. 40–57. Springer International Publishing, Cham (2021). DOI 10.1007/978-3-030-72465-8_3

12. Bergstra, J., Yamins, D., Cox, D.D.: Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In: Proceedings of the 30th International Conference on International Conference on Machine Learning, ICML'13, p. I–115–I–123. JMLR (2013)

13. Blondel, V.D., Decuyper, A., Krings, G.: A survey of results on mobile phone datasets analysis. EPJ Data Science **4**(1), 10 (2015). DOI 10.1140/epjds/s13688-015-0046-0

14. Buckee, C.O., Balsari, S., Chan, J., Crosas, M., Dominici, F., Gasser, U., Grad, Y.H., Grenfell, B., Halloran, M.E., Kraemer, M.U.G., Lipsitch, M., Metcalf, C.J.E., Meyers, L.A., Perkins, T.A., Santillana, M., Scarpino, S.V., Viboud, C., Wesolowski, A., Schroeder, A.: Aggregated mobility data could help fight COVID-19. Science **368**(6487), 145–146 (2020). DOI 10.1126/science.abb8021

15. Carlini, N., Liu, C., Erlingsson, Ú., Kos, J., Song, D.: The secret sharer: Evaluating and testing unintended memorization in neural networks. In: 28th USENIX Security Symposium (USENIX Security 19), pp. 267–284. USENIX Association, Santa Clara, CA (2019)

16. Carlini, N., Tramèr, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, Ú., Oprea, A., Raffel, C.: Extracting training data from large language models. In: 30th USENIX Security Symposium (USENIX Security 21), pp. 2633–2650. USENIX Association (2021)

17. Chaudhuri, K., Monteleoni, C., Sarwate, A.D.: Differentially private empirical risk minimization. Journal of Machine Learning Research **12**(29), 1069–1109 (2011)

18. Chen, S., Fu, A., Shen, J., Yu, S., Wang, H., Sun, H.: RNN-DP: A new differential privacy scheme base on recurrent neural network for dynamic trajectory privacy protection. Journal of Network and Computer Applications **168**, 102736 (2020). DOI 10.1016/j.jnca.2020.102736

19. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. In: NIPS 2014 Workshop on Deep Learning (2014)

20. Dujardin, S., Jacques, D., Steele, J., Linard, C.: Mobile phone data for urban climate change adaptation: Reviewing applications, opportunities and key challenges. Sustainability **12**(4), 1501 (2020). DOI 10.3390/su12041501

21. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Theory of Cryptography, pp. 265–284. Springer Berlin Heidelberg (2006). DOI 10.1007/11681878_14

22. Dwork, C., Roth, A.: The algorithmic foundations of differential privacy. Foundations and Trends® in Theoretical Computer Science **9**(3–4), 211–407 (2014)

23. Eibl, G., Bao, K., Grassal, P.W., Bernau, D., Schmeck, H.: The influence of differential privacy on short term electric load forecasting. Energy Informatics **1**(S1) (2018). DOI 10.1186/s42162-018-0025-3

24. European-Commission: Commission recommendation (eu) 2020/518 of 8 april 2020 on a common union toolbox for the use of technology and data to combat and exit from the COVID-19 crisis, in particular concerning mobile applications and the use of anonymised mobility data. Available online: `https://eur-lex.europa.eu/eli/reco/2020/518/oj` (accessed on 04 July 2021)

25. H. Arcolezi, H., Cerna, S., Couchot, J.f., Guyeux, C., Makhoul, A.: Privacy-preserving prediction of victim's mortality and their need for transportation to health facilities. IEEE Transactions on Industrial Informatics pp. 1–1 (2021). DOI 10.1109/TII.2021.3123588

26. Hewamalage, H., Bergmeir, C., Bandara, K.: Recurrent neural networks for time series forecasting: Current status and future directions. International Journal of Forecasting **37**(1), 388–427 (2021). DOI 10.1016/j.ijforecast.2020.06.008

27. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)

28. Hong, L., Lee, M., Mashhadi, A., Frias-Martinez, V.: Towards understanding communication behavior changes during floods using cell phone data. In: Lecture Notes in Computer Science, pp. 97–107. Springer International Publishing (2018). DOI 10.1007/978-3-030-01159-8_9

29. Imtiaz, S., Horchidan, S.F., Abbas, Z., Arsalan, M., Chaudhry, H.N., Vlassov, V.: Privacy preserving time-series forecasting of user health data streams. In: 2020 IEEE International Conference on Big Data (Big Data). IEEE (2020). DOI 10.1109/bigdata50022.2020.9378186

30. Luca, M., Barlacchi, G., Lepri, B., Pappalardo, L.: A survey on deep learning for human mobility. ACM Comput. Surv. **55**(1) (2021). DOI 10.1145/3485125

31. Mahawaga Arachchige, P.C., Bertok, P., Khalil, I., Liu, D., Camtepe, S., Atiquzzaman, M.: Local differential privacy for deep learning. IEEE Internet of Things Journal **7**(7), 5827–5842 (2020). DOI 10.1109/JIOT.2019.2952146

32. McCandless, D., Evans, T., Quick, M., Hollowood, E., Miles, C., Hampson, D., Geere, D.: World's biggest data breaches & hacks. `https://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/` (2021). Online; accessed 11 March 2021

33. McMahan, H.B., Andrew, G., Erlingsson, U., Chien, S., Mironov, I., Papernot, N., Kairouz, P.: A general approach to adding differential privacy to iterative training procedures. In: Advances in Neural Information Processing Systems (NeurIPS) Workshop on Privacy Preserving Machine Learning (2018)

34. Mir, D.J., Isaacman, S., Caceres, R., Martonosi, M., Wright, R.N.: DP-WHERE: Differentially private modeling of human mobility. In: 2013 IEEE International Conference on Big Data. IEEE (2013). DOI 10.1109/bigdata.2013.6691626

35. de Montjoye, Y.A., Gambs, S., Blondel, V., Canright, G., de Cordes, N., Deletaille, S., Engø-Monsen, K., Garcia-Herranz, M., Kendall, J., Kerry, C., Krings, G., Letouzé, E., Luengo-Oroz, M., Oliver, N., Rocher, L., Rutherford, A., Smoreda, Z., Steele, J., Wetter, E., Pentland, A.".", Bengtsson, L.: On the privacy-conscientious use of mobile phone data. Scientific Data **5**(1), 180286 (2018). DOI 10.1038/sdata.2018.286

36. de Montjoye, Y.A., Hidalgo, C.A., Verleysen, M., Blondel, V.D.: Unique in the crowd: The privacy bounds of human mobility. Scientific Reports **3**(1), 1376 (2013). DOI 10.1038/srep01376

37. Moreno, S.R., da Silva, R.G., Mariani, V.C., dos Santos Coelho, L.: Multi-step wind speed forecasting based on hybrid multi-stage decomposition model and long short-term memory neural network. Energy Conversion and Management **213**, 112869 (2020). DOI 10.1016/j.enconman.2020.112869

38. Murakami, T., Takahashi, K.: Toward evaluating re-identification risks in the local privacy model. Transactions on Data Privacy **14**, 79–116 (2021)

39. Oliver, N., Lepri, B., Sterly, H., Lambiotte, R., Deletaille, S., Nadai, M.D., Letouzé, E., Salah, A.A., Benjamins, R., Cattuto, C., Colizza, V., de Cordes, N., Fraiberger, S.P., Koebe, T., Lehmann, S., Murillo, J., Pentland, A., Pham, P.N., Pivetta, F., Saramäki, J., Scarpino, S.V., Tizzoni, M., Verhulst, S., Vinck, P.: Mobile phone data for informing public health actions across the COVID-19 pandemic life cycle. Science Advances **6**(23), eabc0764 (2020). DOI 10.1126/sciadv.abc0764

40. Orange-Business-Services: Flux vision: real time statistics on mobility patterns (2013). Available online: `https://www.orange-business.com/en/products/flux-vision` (accessed on 01 July 2021)

41. Ouyang, K., Shokri, R., Rosenblum, D.S., Yang, W.: A non-parametric generative model for human trajectories. IJCAI'18, p. 3812–3817. AAAI Press (2018)

42. Pyrgelis, A., Troncoso, C., Cristofaro, E.D.: What does the crowd say about you? evaluating aggregation-based location privacy. Proceedings on Privacy Enhancing Technologies **2017**(4), 156–176 (2017). DOI 10.1515/popets-2017-0043

43. Pyrgelis, A., Troncoso, C., Cristofaro, E.D.: Measuring membership privacy on aggregate location time-series. In: Abstracts of the 2020 SIGMETRICS/Performance Joint International Conference on Measurement and Modeling of Computer Systems, pp. 1–28. ACM (2020). DOI 10.1145/3393691.3394200

44. Rahimi, I., Chen, F., Gandomi, A.H.: A review on COVID-19 forecasting models. Neural Computing and Applications (2021). DOI 10.1007/s00521-020-05626-8

45. Rogers, R., Subramaniam, S., Peng, S., Durfee, D., Lee, S., Kancha, S.K., Sahay, S., Ahammad, P.: Linkedin's audience engagements API: A privacy preserving data analytics system at scale. Journal of Privacy and Confidentiality **11**(3) (2021). DOI 10.29012/jpc.782

46. Schuster, M., Paliwal, K.: Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing **45**(11), 2673–2681 (1997). DOI 10.1109/78.650093

47. Sezer, O.B., Gudelek, M.U., Ozbayoglu, A.M.: Financial time series forecasting with deep learning : A systematic literature review: 2005–2019. Applied Soft Computing **90**, 106181 (2020). DOI 10.1016/j.asoc.2020.106181

48. Shokri, R., Shmatikov, V.: Privacy-preserving deep learning. CCS '15, p. 1310–1321. Association for Computing Machinery, New York, NY, USA (2015). DOI 10.1145/2810103.2813687

49. Shokri, R., Stronati, M., Song, C., Shmatikov, V.: Membership inference attacks against machine learning models. In: 2017 IEEE symposium on security and privacy (SP), pp. 3–18. IEEE (2017). DOI 10.1109/sp.2017.41

50. Song, C., Ristenpart, T., Shmatikov, V.: Machine learning models that remember too much. CCS '17, p. 587–601. Association for Computing Machinery, New York, NY, USA (2017). DOI 10.1145/3133956.3134077

51. Soykan, E.U., Bilgin, Z., Ersoy, M.A., Tomur, E.: Differentially private deep learning for load forecasting on smart grid. In: 2019 IEEE Globecom Workshops (GC Wkshps), pp. 1–6. IEEE (2019). DOI 10.1109/gcwkshps 45667.2019.9024520

52. Stefenon, S.F., Ribeiro, M.H.D.M., Nied, A., Mariani, V.C., dos Santos Coelho, L., da Rocha, D.F.M., Grebogi, R.B., de Barros Ruano, A.E.: Wavelet group method of data handling for fault prediction in electrical power insulators. International Journal of Electrical Power & Energy Systems **123**, 106269 (2020). DOI 10.1016/j.ijepes .2020.106269

53. Tu, Z., Xu, F., Li, Y., Zhang, P., Jin, D.: A new privacy breach: User trajectory recovery from aggregated mobility data. IEEE/ACM Transactions on Networking **26**(3), 1446–1459 (2018). DOI 10.1109/tnet.2018.2829173

54. Vespe, M., Iacus, S.M., Santamaria, C., Sermi, F., Spyratos, S.: On the use of data from multiple mobile network operators in europe to fight COVID-19. Data & Policy **3**, e8 (2021). DOI 10.1017/dap.2021.9

55. Wesolowski, A., Buckee, C.O., Bengtsson, L., Wetter, E., Lu, X., Tatem, A.J.: Commentary: Containing the ebola outbreak - the potential and challenge of mobile network data. PLoS Currents (2014). DOI 10.1371/currents.out breaks.0177e7fcf52217b8b634376e2f3efc5e

56. World-Health-Organization: WHO announces COVID-19 outbreak a pandemic. Available online: `https://www.eu ro.who.int/en/health-topics/health-emergencies/c oronavirus-covid-19/news/news/2020/3/who-annou nces-covid-19-outbreak-a-pandemic` (accessed on 07 September 2020)

57. Xu, F., Tu, Z., Li, Y., Zhang, P., Fu, X., Jin, D.: Trajectory recovery from ASH. In: Proceedings of the 26th International Conference on World Wide Web, pp. 1241–1250. International World Wide Web Conferences Steering Committee (2017). DOI 10.1145/3038912.3052620

58. Yang, Y., Gohari, P., Topcu, U.: On the privacy risks of deploying recurrent neural networks in machine learning. arXiv preprint arXiv:2110.03054 (2021)

59. Yin, M., Sheehan, M., Feygin, S., Paiement, J.F., Pozdnoukhov, A.: A generative model of urban activities from cellular data. IEEE Transactions on Intelligent Transportation Systems **19**(6), 1682–1696 (2018). DOI 10.1109/TITS.2017.2695438

60. Yousefpour, A., Shilov, I., Sablayrolles, A., Testuggine, D., Prasad, K., Malek, M., Nguyen, J., Ghosh, S., Bharadwaj, A., Zhao, J., Cormode, G., Mironov, I.: Opacus: User-friendly differential privacy library in pytorch. In: NeurIPS 2021 Workshop Privacy in Machine Learning (2021)