Generating synthetic mobility data for realistic populations with RNNs to improve utility and privacy

Generating synthetic mobility data for a realistic population with RNNs to improve utility and privacy

Alex Berke, Ronan Doorley, Kent Larson, Esteban Moro
MIT Media Lab, Cambridge, MA, USA
{aberke,doorleyr,kll,emoro}@mit.edu

ABSTRACT

Location data collected from mobile devices represent mobility behaviors at individual and societal levels. These data have important applications ranging from transportation planning to epidemic modeling. However, issues must be overcome to best serve these use cases: The data often represent a limited sample of the population and use of the data jeopardizes privacy.

To address these issues, we present and evaluate a system for generating synthetic mobility data using a deep recurrent neural network (RNN) which is trained on real location data. The system takes a population distribution as input and generates mobility traces for a corresponding synthetic population.

Related generative approaches have not solved the challenges of capturing both the patterns and variability in individuals' mobility behaviors over longer time periods, while also balancing the generation of realistic data with privacy. Our system leverages RNNs' ability to generate complex and novel sequences while retaining patterns from training data. Also, the model introduces randomness used to calibrate the variation between the synthetic and real data at the individual level. This is to both capture variability in human mobility, and protect user privacy.

Location based services (LBS) data from more than 22,700 mobile devices were used in an experimental evaluation across utility and privacy metrics. We show the generated mobility data retain the characteristics of the real data, while varying from the real data at the individual level, and where this amount of variation matches the variation within the real data.

CCS CONCEPTS

• **Computing methodologies** → **Neural networks**; • **Security and privacy**;

KEYWORDS

Synthetic data, recurrent neural networks, mobility, privacy

ACM Reference Format:
Alex Berke, Ronan Doorley, Kent Larson, Esteban Moro. 2022. Generating synthetic mobility data for a realistic population with RNNs to improve utility and privacy. In *The 37th ACM/SIGAPP Symposium on Applied Computing (SAC '22), April 25–29, 2022, Virtual Event.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3477314.3507230

SAC '22, April 25–29, 2022, Virtual Event

ACM ISBN 978-1-4503-8713-2/22/04.
https://doi.org/10.1145/3477314.3507230

1 INTRODUCTION

Location datasets collected from mobile devices have numerous applications ranging from transportation research to analyzing a pandemic [2, 10]. However, these datasets often represent a small sample of the population, limiting their utility. Another important issue is privacy for the device users from whom data were collected, as simple approaches to anonymization are insufficient for spatiotemporal data [9]. Prior works have attempted to mitigate privacy risks with strategies that modify data, yet researchers have shown that risks are still present [12]. Moreover, these modifications decrease data utility.

This work approaches the utility-privacy tradeoff with a system to generate realistic synthetic mobility traces to be used instead of real data. By retaining properties of the real data, the synthetic data can retain utility. And by sufficiently varying from the real data at the individual level, privacy risks can be mitigated. To address the issue of limited sample sizes, the system uses population data as input to generate synthetic data representing that population.

Our approach exploits patterns inherent in location traces by leveraging the success of recurrent neural networks (RNNs) in text generation and modeling our problem similarly. This approach also allows inserting calibrated randomness to manage variation in the model's output. This helps generate data with variation beyond the training data as well as balance the utility-privacy tradeoff.

Contribution: We present a system using an RNN to generate realistic spatiotemporal data representing individuals' mobility over extended periods. The system takes home and work locations as inputs to generate data for a given population size and distribution. Our work includes an experimental implementation, using a location based services (LBS) dataset, that generates data representing individuals' mobility over a 5-day workweek. To evaluate utility we develop and use a variety of metrics that build on previous works. For privacy, we develop metrics to evaluate whether the variation between the synthetic and real data matches the level of variation within the real data, at the individual level.

2 RELATED WORK

Related works using location data from CDRs address issues of limited data by labeling users' data with inferred home and work areas to help expand datasets to match census population estimates [7, 17]. Common approaches use generative algorithms, such as "Exploration and Preferential Return" (EPR) models [13, 17, 19]. These models leverage the predictable nature of human mobility and often assume users are in home and work areas during predefined hours.

The aforementioned works focus on data utility without addressing privacy. Other works use ϵ-differential privacy (DP) [11] in their location data publishing strategies, but without fully addressing

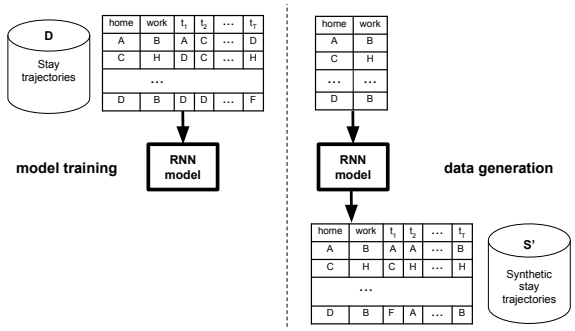

Figure 1: Model training and generation. The RNN is trained with real data, D, where each user's data is a stay trajectory labeled by home, work locations. The trained RNN takes home, work locations as input to generate synthetic data, S'.

our problem with spatiotemporal trajectory data that represents individuals over extended periods. For example [1] use DP to publish aggregate metrics from location data, which [3] noted unsuitable for applications requiring location traces. Other works use DP in generative algorithms. [15] and [16] generate location trajectories with a focus on retaining spatial properties. However, other than being time ordered sequences, data generated by [15] lack temporal information, and [16] is applied to trajectories that are vehicle trips rather than data observed from individuals over a broader space and time. [5] note the challenge of applying DP to sequential data due to its inherent high-dimensionality. They use DP in generating variable length n-grams representing location trajectories. However, their work is limited to short sequences over a small set of discrete locations, such as metro stations [6], and [15] show their approach does not handle spatial data.

Noting limitations of DP, [3] generate spatiotemporal trajectory data to meet alternative privacy criteria, called (k, δ)-plausible deniability. Plausible deniability requires defining a metric, to measure similarity between trajectories, and thresholds, δ and k.

A limitation of these works using DP and plausible deniability is their abstract nature. They provide theory for how, given parameters (ϵ, k, δ), their privacy criteria would be met. But determining parameter values, or analyzing the relationship between these values and privacy for real data, is beyond their scope. The privacy evaluation in this work compares synthetic data to real data.

3 MODELING THE PROBLEM

Similar to related works [15, 16, 19], we transform spatiotemporal data into sequences that discretize time and geographic space. This results in a "stay trajectory" for each user, representing their sequence of visited locations. Sequence indices represent time intervals where values are the location the user stayed for the most time within the interval. We use census areas as locations, and map data points to their containing areas. We represent each stay trajectory as $s = \langle s_1,\ s_2,\ ..., s_T \rangle$ and associate a $\langle home, work \rangle$ pair with each s, where *home* and *work* are areas in s.

3.1 Leveraging a Recurrent Neural Network

RNNs have been successful in generating complex sequences that retain structural properties inherent in text [14]. Stay trajectories have properties similar to text. Both can be represented as sequences of tokens, with temporal and spatial relationships between tokens.

RNNs predict a next element in a sequence conditioned on previous elements. Each prediction step samples from a distribution of candidate next elements, where this process allows parameterizing randomness for the model's output. By feeding a model's predictions back to itself, novel sequences can be generated.

To leverage RNNs for our use case we prefix each s with its $\langle home, work \rangle$ pair, $\langle home, work, s_1, s_2, ..., s_T \rangle$. The prefixes serve as labels and the prefixed trajectories are used to train the model. The model can then learn relationships between the prefixes and the tokens that follow, such as how tokens in the *home*, *work* prefix positions are likely candidates for nighttime and workday hours, along with other structural relationships between tokens.

For data generation, we feed $\langle home, work \rangle$ pairs to the trained model as labels for it to generate corresponding stay trajectories. The RNN treats the input $\langle home, work \rangle$ pairs as prefixes for sequences it learned to complete. The sequences it then generates are the synthetic stay trajectories with the given $\langle home, work \rangle$ labels.

4 EXPERIMENTAL EVALUATION

The code for the work described in this section is open source[1].

4.1 Data Panel and Preprocessing

4.1.1 Data Panel. An LBS dataset was provided by a location intelligence company. It was collected from users who opted-in to share data anonymously through a GDPR-compliant framework. It was provided as table rows containing a device ID, geolocation coordinates, timestamp, and estimated time the device was in the location. We created a panel from the first 5-day workweek of May 2018, restricted to data points reported within the 3 counties surrounding Boston, MA. After filtering, the panel included 22,707 devices that each reported at least 3 days and nights of data in the 5-day period.

4.1.2 Home and Work. We defined functions, $inferHome$ and $inferWork$, that take a stay trajectory as input and return census tracts for *home*, *work*. These were used to label stay trajectories and evaluate model output. The area the user stayed most 8pm to 9am is inferred as *home* and the area stayed most during the remaining hours is *work*[2]. We applied the $inferHome$ function to the data panel to result in corresponding census tract level population estimates. Compared to ACS 2018 census estimates [4] there is a Pearson correlation coefficient of 0.648. This relatively high correlation helps validate methods and shows data representativeness[3].

4.1.3 Data Used for Model Training, Generation and Evaluation. Stay trajectories were created for each device in the panel with time intervals of 1 hour and census tracts as areas. These parameters were chosen based on panel size and data sparsity; with more

[1] https://github.com/aberke/lbs-data/blob/master/trajectory_synthesis
[2] What we call *work* can be considered any secondary location to *home*.
[3] For comparison we used data from location data company Safegraph. They made statistics from their September 2019 data available, including the number of devices residing in each census area. We measured the correlation between their device populations and census estimates at the census tract level, restricting analysis to the geographic region of our study. This included 396,061 devices. The Pearson correlation is 0.122. For details see https://github.com/aberke/lbs-data/blob/master/safegraph-comparison.ipynb.

Embedding size	Dimension of the embedding layer.	128
Layer size	Number of LSTM units in each hidden layer.	128
Layers	Number of hidden layers.	6
Dropout	Rate at which weighted connections between units are randomly excluded during training (regularization).	0.1
Maximum length	Max number of previous sequence tokens used to predict the next token (length needed to learn patterns).	60

Table 1: RNN model hyperparameters.

metric	synthetic sample	real secondary sample	randomly generated sample
trip distance (KL divergence)	0.0008	0.0015	0.3655
locations per user (KL divergence)	0.0124	0.0044	-2.4587
aggr. time per location (KL divergence)	0.0366	0.0085	0.9608
home label error rate	0.1375	0.0863*	0.9995
work label error rate	0.2675	0.2415*	0.9235

Table 2: Utility metrics evaluating error from real data. Two baselines are used for comparison: A secondary real data sample and randomly generated sample. Lower values indicate lower error. *Measured as the rate at which labels change between weeks (section 4.3.4).

data, greater spatial and temporal precision may be used. Stay trajectories were then prefixed with their inferred $\langle home, work \rangle$ label.

D: 22,707 stay trajectories (from panel). D is used to train the model.
S: A subset of 2000 stay trajectories randomly sampled from D.
S': 2000 synthetic stay trajectories where the distribution of $\langle home, work \rangle$ label pairs is consistent with S'. S' is generated by providing the $\langle home, work \rangle$ label pair for each $s \in S$ as model input.

4.2 RNN Model

The model was implemented with the textgenrnn library [21]. At a high level, its architecture can be described as follows. An input layer is followed by an embedding layer, then by "hidden" layers of LSTM units, then by an attention layer, then the output layer, where the embedding and LSTM layers are each skip-connected to the attention layer. Numerous models were trained with different hyperparameters. Their outputs were evaluated with metrics described in sections 4.3 and 4.4 to select a best model. These selected hyperparameters are shown in Table 1. The epoch and batch size were 50 and 1024, respectively, and a "temperature" value of 1 parameterized randomness in the predictive sampling step. The following sections report on the output of this model.

4.3 Utility Evaluation

We build on metrics from previous works [3, 17, 19] and borrow their evaluation strategy of using Kullback-Leibler divergence [8] to compute differences between metric distributions for the real versus synthetic data. Results are shown in Table 2 and Figure 2. We also evaluate how well synthetic trajectories match their input labels. Note we cannot compare metrics between works because the evaluations are for different data generation processes and use different datasets. Even studies that apply the same processes to multiple datasets yield different results for each dataset [15, 19]. To help evaluate utility, we follow the methods of [3] to create baselines by drawing a secondary real sample and generating a random sample, where each sample matches in size ($|S|$=2000) and $\langle home, work \rangle$ label distribution.

Similar metrics between the synthetic and real samples may imply the synthetic generation process is similar to drawing from the real data in terms of utility.

4.3.1 Trip Distances. We measure trip distances as the line distance between area centroids, counted when consecutive areas in stay trajectories differ. Distance distributions are transformed into discrete probability distributions, $P(d)$, via a histogram. Distributions between the synthetic and real data closely match. The distribution for the randomly generated sample is a different shape, showing this is not simply due to the distribution of distances between areas.

4.3.2 Locations Per User. EPR modeling works have used this metric to describe user "exploration" and evaluate synthetic data [17, 19]. Following their methods, L is locations per user and we compute the distribution of locations per user, $P(L)$.

4.3.3 Proportion of Aggregate Time Spent Per Location. We compute the aggregate time spent in each area for each sample. Comparison is then made to S rather than D because the distribution of $\langle home, work \rangle$ pairs is then consistent and where users spend time is biased to where they live and work.

4.3.4 Home, Work Label Error. For each input label $\langle home_i, work_i \rangle$ and output $s'_i \in S'$, we count errors when $home_i \neq inferHome(s'_i)$ and $work_i \neq inferWork(s'_i)$, and calculate error rate as total errors over |S'|. For a real baseline, we draw a second panel using the same criteria for D but for the following week. Error is then calculated as the rate inferred labels changed for users between weeks.

4.4 Privacy Evaluation

Section 2 notes the challenges of applying DP to spatiotemporal trajectory data, and how related works have limited their application of DP to different problems or used plausible deniability [3]. We build on these previous works. Our evaluation considers S' as an alternative to S, sampled from D. We measure the similarity between any synthetic trajectory in S' and any real trajectory in D, and check they are not too similar, when compared to similarities between real trajectories in S and any other real trajectories in D.

For any two trajectories, s_i, s_j, we measure the difference between them, $d(s_i, s_j)$, as the Levenshtein edit distance [18]. We compute the minimum distance between a given s and any other s_j in D, which we call $min\text{-}dist(s, D)$.

$$min\text{-}dist(s, D) = d(s, s_j)\ s.t. \forall\ s_j, s_k \in D, d(s, s_j) \leq d(s, s_k)$$

These values are computed for each $s \in S \subset D$, but where direct comparison of s to itself is avoided. Similarly, for each $s' \in S'$,

$$min\text{-}dist(s', D) = d(s', s_j)\ s.t. \forall\ s_j, s_k \in D, d(s', s_j) \leq d(s', s_k)$$

Our evaluation then considers, for any distance m,

$$Pr[min\text{-}dist(s', D) \leq m] \leq Pr[min\text{-}dist(s, D) \leq m]$$

Probabilities describe the evaluation since sampling S and generating S' are stochastic processes. For our experiment, we evaluate empirical distributions. i.e. $Pr[min\text{-}dist(s, D) \leq m]$ is estimated as the proportion of $min\text{-}dist(s, D)$ values where $min\text{-}dist(s, D) \leq m$.

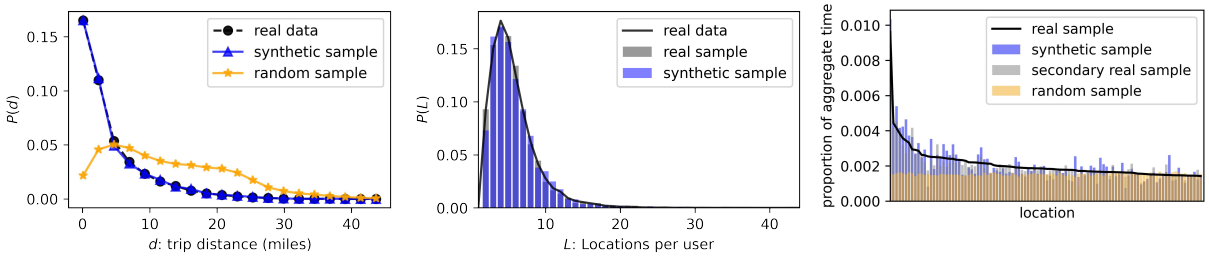

Figure 2: Utility metrics. (Left) Distribution of trip distances. (Center) Distribution of locations per user. The distribution for the randomly generated sample is centered beyond outliers in the real data and not shown. (Right) Proportion of aggregate time spent in each location. Locations are sorted by aggregate time for the real sample, and shown for the top-100 locations.

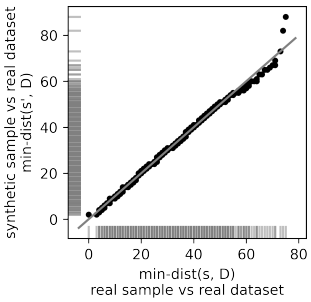

Figure 3: Q-Q plot comparing *min-dist* values.

To help evaluate results over the range of m values, we use Q-Q plots (Figure 3). The distributions of *min-dist* values are sorted and the Q-Q plot matches corresponding m values for the S and S' against each other, with values for S and S' on the x and y axes, respectively. A 45-degree line represents matching distributions, and points on or above the line represent where the privacy evaluation is satisfied. Values closer to the origin are more important, as these are for smaller *min-dist* values, where privacy risk is higher.

Experiment results nearly track the 45-degree line, implying the model generates synthetic data that differs from the real data nearly as much as the real data differs from itself, at the individual level.

5 CONCLUSION AND FUTURE WORK

This work evaluated synthetic data as an alternative to real data, to offer similar utility while mitigating privacy risks. For evaluation, a synthetic data sample with home, work labels matching the distribution of a real sample was used. Yet since the system generates data with variation, it may be used to generate data for much larger populations, such as a population based on census data.

Our experiment used LBS data with methods that future work can extend to other forms of location data. Future work can also test combining various data sources in model training to avoid common problems of de-duplicating data [20].

REFERENCES

[1] Gergely Acs and Claude Castelluccia. 2014. 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*. 1679–1688.

[2] Hugo Barbosa, Marc Barthelemy, Gourab Ghoshal, Charlotte R James, Maxime Lenormand, Thomas Louail, Ronaldo Menezes, José J Ramasco, Filippo Simini, and Marcello Tomasini. 2018. Human mobility: Models and applications. *Physics Reports* 734 (2018), 1–74.

[3] Vincent Bindschaedler and Reza Shokri. 2016. Synthesizing plausible privacy-preserving location traces. In *2016 IEEE Symposium on Security and Privacy (SP)*. IEEE, 546–563.

[4] U.S. Census Bureau. 2019. American Community Survey 2014-2018 5-year Estimates. https://data.census.gov/cedsci/table.

[5] Rui Chen, Gergely Acs, and Claude Castelluccia. 2012. Differentially private sequential data publication via variable-length n-grams. In *Proceedings of the 2012 ACM conference on Computer and communications security*. 638–649.

[6] Rui Chen, Benjamin CM Fung, Bipin C Desai, and Nériah M Sossou. 2012. Differentially private transit data publication: a case study on the montreal transportation system. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 213–221.

[7] Serdar Çolak, Lauren P Alexander, Bernardo G Alvim, Shomik R Mehndiratta, and Marta C González. 2015. Analyzing cell phone location data for urban travel: current methods, limitations, and opportunities. *Transportation Research Record* 2526, 1 (2015), 126–135.

[8] Thomas M Cover. 1999. *Elements of information theory*. John Wiley & Sons.

[9] Yves-Alexandre De Montjoye, César A Hidalgo, Michel Verleysen, and Vincent D Blondel. 2013. Unique in the crowd: The privacy bounds of human mobility. *Scientific reports* 3 (2013), 1376.

[10] Ronan Doorley, Alex Berke, Ariel Noyman, Luis Alonso, Josep Frriz Ribo, Vanesa Arroyo, Marc Pons, and Kent Larson. 2021. Mobility and COVID-19 in Andorra: Country-scale analysis of high-resolution mobility patterns and infection spread. *IEEE Journal of Biomedical and Health Informatics* (2021), 1–1. https://doi.org/10.1109/JBHI.2021.3121165

[11] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*. Springer, 265–284.

[12] Marco Fiore, Panagiota Katsikouli, Elli Zavou, Mathieu Cunche, Françoise Fessant, Dominique Le Hello, Ulrich Aivodji, Baptiste Olivier, Tony Quertier, and Razvan Stanica. 2020. Privacy in trajectory micro-data publishing: a survey. *Transactions on Data Privacy* 13 (2020), 91–149.

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

[14] Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850* (2013).

[15] Mehmet Emre Gursoy, Ling Liu, Stacey Truex, and Lei Yu. 2018. Differentially private and utility preserving publication of trajectory data. *IEEE Transactions on Mobile Computing* 18, 10 (2018), 2315–2329.

[16] Xi He, Graham Cormode, Ashwin Machanavajjhala, Cecilia M Procopiuc, and Divesh Srivastava. 2015. DPT: differentially private trajectory synthesis using hierarchical reference systems. *Proceedings of the VLDB Endowment* 8, 11 (2015), 1154–1165.

[17] Shan Jiang, Yingxiang Yang, Siddharth Gupta, Daniele Veneziano, Shounak Athavale, and Marta C González. 2016. The TimeGeo modeling framework for urban mobility without travel surveys. *Proceedings of the National Academy of Sciences* 113, 37 (2016), E5370–E5378.

[18] Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, Vol. 10. 707–710.

[19] Luca Pappalardo and Filippo Simini. 2018. Data-driven generation of spatio-temporal routines in human mobility. *Data Mining and Knowledge Discovery* 32, 3 (2018), 787–829.

[20] Feilong Wang, Jingxing Wang, Jinzhou Cao, Cynthia Chen, and Xuegang Jeff Ban. 2019. Extracting trips from multi-sourced data for mobility pattern analysis: An app-based data example. *Transportation Research Part C: Emerging Technologies* 105 (2019), 183–202.

[21] Max Woolf. 2019. textgenrnn. https://github.com/minimaxir/textgenrnn