Generating Synthetic Mobility Traffic Using RNNs

Vaibhav Kulkarni
Distributed Object Programming Lab
University of Lausanne, Switzerland
Vaibhav.Kulkarni@unil.ch

Benoît Garbinato
Distributed Object Programming Lab
University of Lausanne, Switzerland
Benoit.Garbinato@unil.ch

ABSTRACT

Mobility trajectory datasets are fundamental for system evaluation and experimental reproducibility. Privacy concerns today, however, have restricted sharing of such datasets. This has led to the development of synthetic traffic generators, which simulate moving entities to create pseudo-realistic trajectory datasets. Existing work on traffic generation, superficially matches *a-priori* modeled mobility characteristics, which lacks realism and does not capture the substantive properties of human mobility. Critical applications however, require data that contains these complex, candid and hidden mobility patterns. To this end, we investigate the effectiveness of Recurrent Neural Networks (RNN) to learn these hidden patterns contained in an original dataset to produce a realistic synthetic dataset. We observe that, the ability of RNNs to learn and model problems over sequential data having long-term temporal dependencies is ideal for capturing the inherent properties of location traces. Additionally, the lack of intuitive high-level spatiotemporal structure and instability, guarantees trajectories that are different from the ones seen in the training dataset. Our preliminary evaluation results show that, our model effectively captures the sleep cycles and stay-points commonly observed in the considered training dataset, along with preserving the statistical characteristics and probability distributions of the movement transitions. Although, many questions remain to be answered, we show that generating synthetic traffic by learning the innate structure of human mobility through RNNs is a promising approach.

CCS CONCEPTS

• **Computing methodologies → Sequential decision making**;

KEYWORDS

Synthetic mobility traffic; Mobility behavior; Recurrent neural networks

ACM Reference Format:
Vaibhav Kulkarni and Benoît Garbinato. 2017. Generating Synthetic Mobility Traffic Using RNNs. In *First ACM SIGSPATIAL Workshop on Artificial Intelligence and Deep Learning for Geographic Knowledge Discovery , November 7–10, 2017, Los Angeles Area, CA, USA.* 4 pages. https://doi.org/10.1145/3149808.3149809

GeoAI'17, November 7–10, 2017, Los Angeles Area, CA, USA

ACM ISBN 978-1-4503-5498-1/17/11…$15.00
https://doi.org/10.1145/3149808.3149809

1 INTRODUCTION

The pervasiveness of mobile devices equipped with internet connectivity and global-positioning (GPS) functionality has resulted in the collection of large volumes of mobility trajectory data of individuals. This data is used for a variety of applications such as designing and evaluating systems aimed at mobility prediction, urban planning, consumer profiling and traffic management. However, sharing such datasets with untrusted third parties have several privacy implications. Simple heuristics can be applied on such datasets to derive personally identifiable information (PII) of users for blackmailing or stalking purposes [11]. Furthermore, data breaches, unlawful data exchanges and security vulnerabilities has restricted sharing of such datasets for development and research purposes.

This has led to the usage of synthetic traffic generators that simulate or mimic the behavior of moving entities. However, existing traffic generators rely on deterministic models having predetermined movement distribution, which fails to capture the behavioral realism. The Brinkoff data generator [2] uses the road network and a perturbation model to generate mobility traces. The BerlinMod Traffic Generator [3] relies on the Berlin road network and the Secondo DBMS [1] to generate data. MNTG traffic generator [14] provides a web-based road network trajectory generator, which is based on Brinkhoff and BerlinMod movement models. A recent trajectory generator, called Hermoupolis [16], uses existing trajectory datasets to generate a larger synthetic dataset. The underlying idea of this model is to extract semantic data from the raw data which is then used to construct a realistic user behavioral model. To generate synthetic data, such generators pick new sets of semantics and use the extracted mobility behaviors to generate trajectories. Such approaches result in purely discriminative behavioral models, i.e., the conditional probability distribution of a data point is learnt according to another point. Although, such models are suitable for generating datasets that address use cases such as trajectory indexing, they lack realism specially in capturing human behavior, which expands beyond modeling the semantics, trip-based and trajectory-based movements. For example, a user can showcase complex routes to travel from point A to B, vary the wake up-sleep and weekend cycle or change behavior to visit some places depending on certain external factors. These changes are critical for applications such as consumer profiling and behavioral analysis.

To this end, we present a synthetic traffic generator that uses machine learning, i.e., recurrent neural networks (RNN) for extracting the substantive behavioral patterns of users from actual datasets. We then use this trained model to create new and larger datasets, characterized by features that resemble the true properties of users from an actual dataset. Our approach combines the discriminative model with the generative model to learn the joint

[1] Secondo DBMS: http://dna.fernuni-hagen.de/secondo/

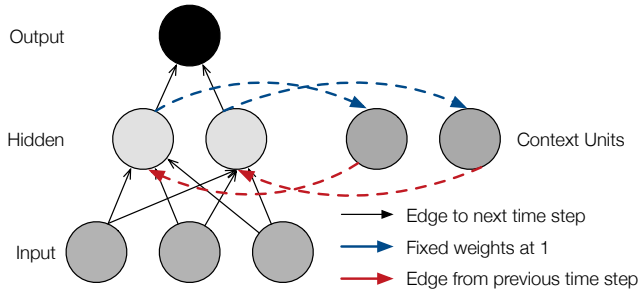


Figure 1: A recurrent neural network architecture. Hidden layer is connected to the context units which feeds back into the hidden layer at the subsequent time step.

probability distribution of a dataset. Training in this manner, restricts the output trajectories to the bounds derived from the dataset, however, the generative model results in producing new trajectory sets. In addition to generating synthetic traffic, the trained model can be used to capture the generalized mobility patterns in a given area, which may include the frequently visited places, commonly followed trajectories and transportation modes utilized.

The rest of this paper is organized as follows. Section 2 presents the necessary background knowledge about RNNs to illustrate our system design. The related literature and the associated shortcomings are discussed in Section 3. We present our system model in Section 4. The evaluation results are presented in Section 5. Finally, the conclusion and future work is presented in Section 6.

2 RECURRENT NEURAL NETWORKS

A typical feed forward neural network has connections from layer n to layer $n + 1$. A key determinant, differentiating RNN is the connection from layer n to layer n in addition to the regular connections as shown in Figure 1. Such loops enable the network to compute on data from previous cycles, creating a network memory. This influences the network predictions to be influenced by the past values making it ideal to learn a sequence. The length of the network memory is not indefinite and gradually degrades with older information being less relevant. A drawback of RNN is that it suffers from the vanishing gradient problem, which also hinders remembering the past inputs. In order to address this, long short term memory (LSTM) is used which also bridges the long time lags between the inputs. This capability is used to learn sequential data using the recurrent connections between their neural activations at consecutive time steps. For a given input x_t at a time step t the network creates a hidden state h_t, such that it is a non-linear function of the previous hidden state h_{t-1}.

RNNs read through a sequence iteratively, preserving the structure in the model. It goes through each element of the sequence and updates its representation based on that item and the input from the previous state. At each time-step n, the number of hidden unit dimensional vector represents the input sequence. The connections between the hidden units and their respective projections are preserved making learning tractable. Gating units are often utilized in a RNN model, to transform the information flow in a more structured manner. They also control the proportion of the past information which should go forward and models the network to adaptively forget information. The model's remembrance is controlled by averaging the previous hidden state output with the current output. A crucial factor determining RNN suitability is the dataset size, as RNNs have poor generalization properties on small dataset volumes [12].

The effectiveness of RNN to learn the patterns on sequential data has been successfully applied for text generation [15], image captioning [13] and action recognition [17]. Sutskever et al. [18] showed that, when provided with a limited set of vocabulary, RNN outperforms a machine translation system with an unlimited vocabulary on a large scale machine translation task. Our motivation to employ LSTM-based RNN to generate mobility trajectories without making any assumptions regarding the problem structure is based on the successful application of RNN for sequence learning problems.

3 RELATED WORK

A majority of the existing techniques to generate synthetic traces are based on trajectory modeling. Jian et al. [8] characterizes mobility as goal oriented in terms of Levy flight behavior attributed to the underlying street network. Here, trajectory is modeled in terms of several flights between the underlying street network, with purposive destination locations. The effect of space on the movement patterns is taken into account by modeling the length and the frequency of a flight according to the power-law distribution. Ghosh et al. [4] model trajectories by finding correlation between user-place, place-place and user-user from the GPS traces. These correlations are then used to form a temporal node-based graph structure for user's trajectory. Kim et al. [9] model mobility, characterized by the speed and pause time of the movements, which follow a log-normal distribution. Using the above techniques, the problem of modeling human mobility becomes quite challenging for unpredictable behaviors.

Another category of techniques rely on well established mobility models of moving objects. Random walk is one of such approaches in which the path of a mathematical object is modeled as a succession of random steps on an arbitrary mathematical space. In the case of random waypoint and random direction model, the movement of mobile users is characterized according to the changes in their location/direction depending on random changes in velocity and acceleration over time. In the truncated Levy walk model, mobility is modeled to follow truncated power-law and further constrained to geographical features such as walk boundary, obstructions and traffic. In the above techniques, the mobile entities can stop suddenly or turn very sharply, failing to capture the true movement patterns of mobile objects. To eliminate such behaviors, Jean-Daniel et al. propose Gauss-Markov (GM) mobility model [1] , to limit the sudden stops and turns within specific regions. In the reference point group mobility model (RPMG) [7], the relationships between different mobile objects is considered to generate synthetic traces as a group of entities. In the above models, the speed and the direction of movement at a new timestamp has no relation to the past locations, furthermore the mobility models are based on stochastic processes and do not truly reflect the realistic mobility characteristics. Furthermore, these approaches result in creation of mobile objects at the same locations in a periodic manner due to the use of bounding parameters.

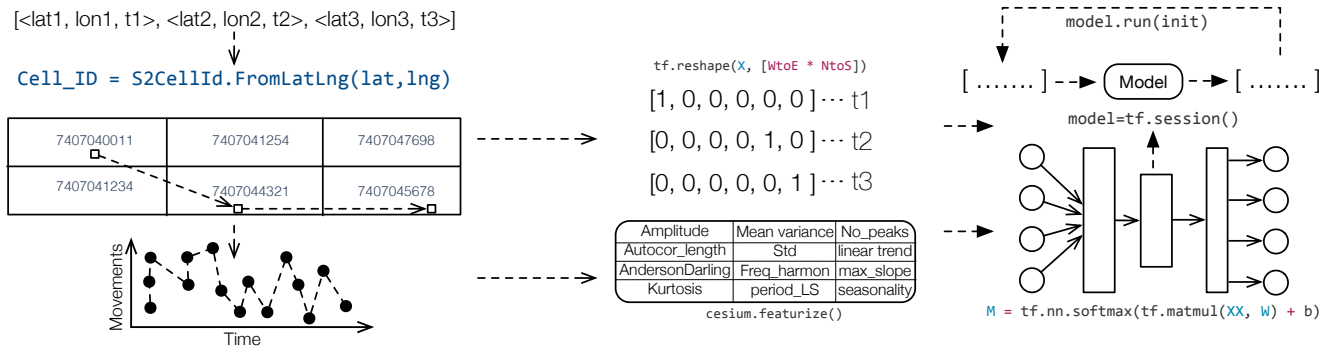


Figure 2: Model training and trajectory generation. The coordinates are mapped on to a grid which are than one hot encoded. Feature exploration is performed on the discretized movements. The extracted features and the vectors are then used for training. The formulated model is instantiated with a sequence of grids to initiate the trajectory generation phase.

4 SYSTEM MODEL

Problem Statement: Given a dataset of trajectories belonging to actual moving subjects, $t_r = \langle u_1, u_2 ... u_n \rangle$, such that $u_i = \langle ..., s_i ... \rangle$ is a trajectory of a user u_i and each point s_i is a three item tuple, (lat_i, lon_i, t_i), where lat and lon are the latitude and longitude coordinates and t is the timestamp, our goal is to extract and learn the user mobility behaviors, in a way that facilitates generation of synthetic but realistic mobility traffic data of a fictional subject $f_i = \langle ..., fs_i ... \rangle$ such that fs_i is also a three item tuple containing the latitude, longitude and timestamp generated by the trained model.

System Design: We base our system design on Long Short-term Memory(LSTM) recurrent neural networks(RNN), by configuring the network model to learn convoluted sequences and extend it to formulate predictions in the spatiotemporal domain [5]. The network with an attention mechanism is trained using actual user trajectories to make it deterministic and follow road networks or other valid paths taken by the legitimate users. Training the model by shuffling the user trajectories, incorporates stochasticity and fuzziness in the model [6], impacting its probabilistic output distribution, which leads to generation of novel trajectory sets. Finally, the complete trajectory sequences can be generated by iteratively feeding the current output trajectory sequence as input to the next step to the trained model, starting at some arbitrary location.

Implementation: Our implementation is based on the TensorFlow library[2] and is depicted in Figure 2. In order to construct the model, we first discretize the space by mapping the coordinates to grids by using the Google S2 library[3] for dimensionality reduction. We configure the S2 library to map each coordinate pair to a cell of dimension $38m^2$. This choice is motivated by the localization accuracy of a typical GPS sensor and the performance complexity involved when subdividing the cells to the leaf level. The cell ID's and the timestamp's are one hot encoded and the resulting vectors are fed as inputs to the network. The network is updated at every instant which enables the next movements to be dependent on the recently seen inputs. In order to bound the outputs, we extract features from the input trajectories which ensures that the movement properties are preserved in the synthetically generated traces. Some of the features include the amplitude of movement which captures the difference between the minimum and maximum magnitude of the movements, the autocorrelation length which captures the periodicity, mean variance etc. In order to compute the features, we use the Cesium library[4], which is used to featurize time-series data.

Challenges: We observe that, batch training such a network, with a dataset of around 191 users for roughly 10,000 epochs is sufficient to capture the basic pattern of mobility behavior including the sleep and wake up cycles and the visitation patterns amongst commonly visited places in the area under consideration. However, we observe that, the model does not preserve the ordering of the commonly visited places and the associated transitions between them. We argue that, it can be addressed by selecting the appropriate features which preserve the structure of the trajectories. Training the model with fine-grained features can eliminate such problems. Along with the above issue, the training process presents some interesting challenges, mainly in selecting the size of the network, amount of memory, dealing with the instability while generating trajectories, amount of noise to be injected to increase the models robustness and bounding the outputs by the properties of the real users. We will address such aspects in our future work.

5 EVALUATION

The model training and evaluation results are based on the Nokia Mobile Dataset [10]. It consists of mobility traces of 191 users collected in Switzerland over a period of two years. We first examine the matching of the generated traces to the road network with respect to the number of trajectories and training epochs. As seen in Figure 3, after 60,000 epochs the model learns the paths typically adopted by the moving objects and starts replicating it. We also observe that, the model learns the common points where the objects stop and the stoppage durations while moving.

Next, we evaluate the prediction accuracy of the network. In order to compute it, we first extract the most frequent transitions between the hotspots in the dataset. We observe that, as the number of users considered in the training phase increases, the models next place prediction accuracy increases. However, on the contrary, the prediction accuracy of the next trajectory decreases. This is crucial to validate the generalization property of the model over

[2]TensorFlow: www.tensorflow.org
[3]Google S2: pypi.python.org/pypi/s2sphere/
[4]Cesium: cesium-ml.org

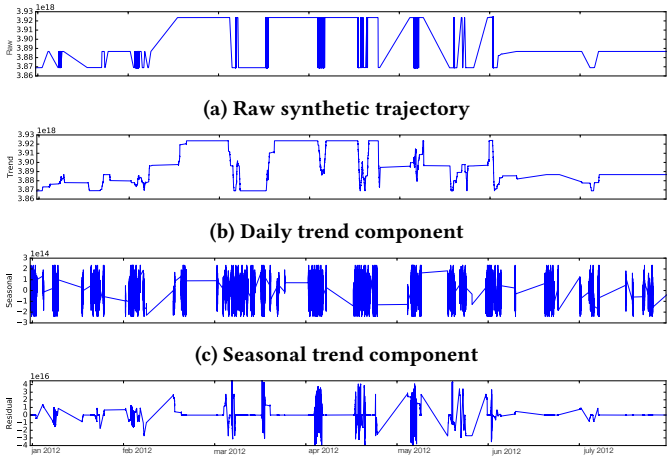
Figure 3: Road network matching accuracy with respect to the the number of users and the number of training epochs.

the training dataset. Our future work will adopt metrics such as negative log-likelihood to quantify such accuracies.

Further, to validate the similarity of the trajectories to the actual user behaviors, we compute the same features used to train the model on the generated traces. We observe that, the model is able to learn the sleep and wake up cycles, movement periodicity and the variance in the movement distance magnitudes. We also calculate the season trend decomposition of a generated trace as shown in Figure 4. Although, the model preserves the weekday and weekend patterns, we observe some gaps in trajectories and some incoherent movement transitions. We argue that such abnormalities can be addressed by using a Bi-directional RNN and echo state network (ESN). The applicability of such mechanisms and validating the generated trajectories with additional mobility quantifiers such as movement entropy, number of visited places, visitation frequency, evolution in the visited places with time, etc. will be addressed in our future work. We will also validate the generated data points by performing nonlinear logistic regression to discriminate between the actual and synthetic data [6].

6 CONCLUSION

In this paper, we have highlighted the necessity of devising generative models for traffic data generation that outreach the limitations of the existing discriminative models. Our proposed approach uses a long short-term memory based recurrent neural network for generating new data on the basis of an existing dataset. The generated data is suitable for uses cases such as mobility prediction and behavioral analysis. Our future work will focus on evaluating the statistical validity of the generated data and on the challenge of transposing the learnt behavioral model to new areas by applying transfer learning. Although the preliminary results are promising, many questions remain to be answered such as; (1) relationship between the network model and dataset realism, dataset magnitude and realism (2) mapping the knowledge from a known region to another unknown (target) region and use this knowledge to categorize the users in the target region (3) privacy measures to prevent backtracking of original mobility traces or reproducing the mobility pattern of any individual (4) quantifying the realism in generated traces. In order to add stochasticity in the training data, a part of our future work is to collect data from sources which do not have predictable/repetitive behavior. Our target subjects include students on



(a) Raw synthetic trajectory

(b) Daily trend component

(c) Seasonal trend component

(d) Periodic trend component

Figure 4: Seasonal trend decomposition of the synthetic trajectory.

an exchange program and short-term tourists. We believe that such measures can address the problem of generating datasets having homogenous distribution.

Acknowledgement: This work is partially supported by the Swiss National Science Foundation grant 157160.

REFERENCES

[1] J. D. M. M. Biomo, T. Kunz, and M. St-Hilaire. An enhanced gauss-markov mobility model for simulations of unmanned aerial ad hoc networks. In *2014 7th IFIP Wireless and Mobile Networking Conference (WMNC)*, pages 1–8, May 2014.
[2] T. Brinkhoff. Generating network-based moving objects. In *Scientific and Statistical Database Management, 2000. Proceedings. 12th International Conference on*, pages 253–255. IEEE, 2000.
[3] C. Düntgen, T. Behr, and R. H. Güting. Berlinmod: a benchmark for moving object databases. *The VLDB JournalâĂŤThe International Journal on Very Large Data Bases*, 18(6):1335–1368, 2009.
[4] S. Ghosh and S. K. Ghosh. Modeling of human movement behavioral knowledge from gps traces for categorizing mobile users. In *WWW*, 2017.
[5] A. Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.
[6] M. Gutmann and A. Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*, volume 1, page 6, 2010.
[7] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang. A group mobility model for ad hoc wireless networks. In *MSWiM '99*, 1999.
[8] B. Jiang, J. Yin, and S. Zhao. Characterizing the human mobility pattern in a large street network. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 80 2 Pt 1:021136, 2009.
[9] M. Kim, D. Kotz, and S. Kim. Extracting a mobility model from real user traces. *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*, pages 1–13, 2006.
[10] N. Kiukkonen, J. Blom, O. Dousse, D. Gatica-Perez, and J. Laurila. Towards rich mobile phone datasets: Lausanne data collection campaign. *Proc. ICPS, Berlin*, 2010.
[11] J. Krumm. Inference attacks on location tracks. In *Pervasive*, 2007.
[12] Z. C. Lipton. A critical review of recurrent neural networks for sequence learning. *CoRR*, abs/1506.00019, 2015.
[13] J. Mao, W. Xu, Y. Yang, J. Wang, and A. L. Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). *CoRR*, abs/1412.6632, 2014.
[14] M. F. Mokbel, L. Alarabi, J. Bao, A. Eldawy, A. Magdy, M. Sarwat, E. Waytas, and S. Yackel. Mntg: an extensible web-based traffic generator. In *International Symposium on Spatial and Temporal Databases*, pages 38–55. Springer, 2013.
[15] R. Nallapati, B. Zhou, C. N. dos Santos, and y. ĂĞaglar GülÃğehre and Bing Xiang, booktitle=CoNLL. Abstractive text summarization using sequence-to-sequence rnns and beyond.
[16] N. Pelekis, S. Sideridis, P. Tampakis, and Y. Theodoridis. Hermoupolis: a semantic trajectory generator in the data science era. *SIGSPATIAL Special*, 7(1):19–26, 2015.
[17] S. Sharma, R. Kiros, and R. Salakhutdinov. Action recognition using visual attention. *CoRR*, abs/1511.04119, 2015.
[18] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014.