Practical Prediction of Human Movements Across Device Types and Spatiotemporal Granularities

Babak Alipour* babak.ap@ufl.edu
Leonardo Tonetto† tonetto@in.tum.de
Roozbeh Ketabi* roozbeh@ufl.edu
Aaron Yi Ding‡ aaron.ding@tudelft.nl
Jörg Ott† ott@in.tum.de
Ahmed Helmy* helmy@ufl.edu

* University of Florida, USA † Technical University of Munich, Germany ‡ TU Delft, The Netherlands

***Abstract*—Understanding and predicting mobility are essential for the design and evaluation of future mobile edge caching and networking. Consequently, research on prediction of human mobility has drawn significant attention in the last decade. Employing information-theoretic concepts and machine learning methods, earlier research has shown evidence that human behavior can be highly predictable.**

Despite existing studies, more investigations are needed to capture intrinsic mobility characteristics constraining predictability, and to explore more dimensions (e.g. device types) and spatio-temporal granularities, especially with the change in human behavior and technology. We analyze extensive longitudinal datasets with fine spatial granularity (AP level) covering 16 months. The study reveals *device type* as an important factor affecting predictability. Ultra-portable devices such as smartphones have "on-the-go" mode of usage (and hence dubbed "*Flutes*"), whereas laptops are "sit-to-use" (dubbed "*Cellos*").

The goal of this study is to investigate practical prediction mechanisms to quantify predictability as an aspect of human mobility modeling, across time, space and *device types*. We apply our systematic analysis to wireless traces from a large university campus. We compare several algorithms using varying degrees of temporal and spatial granularity for the two modes of devices; *Flutes* vs. *Cellos*. Through our analysis, we quantify how the mobility of *Flutes* is less predictable than the mobility of *Cellos*. In addition, this pattern is consistent across various spatio-temporal granularities, and for different methods (Markov chains, neural networks/deep learning, entropy-based estimators). This work substantiates the importance of predictability as an essential aspect of human mobility, with direct application in predictive caching, user behavior modeling and mobility simulations.

I. Introduction & Related work

In recent years, large-scale research on human mobility has thrived due to the availability of location data collected from portable computing and communication devices, such as laptops, smartphones, smartwatches and fitness trackers. One particular aspect of human mobility that has gained a lot of attention lately is predictability. Prediction techniques constitute fundamental mechanistic building blocks for many mobile protocols and applications, ranging from resource allocation to caching and recommender systems, among others [1], [2].

The seminal work by [3], utilizing cellular network data, established an approach towards understanding and measuring predictability of human mobility patterns, with their equally important contribution with respect to the data-driven analysis of large mobile populations, and their efforts in devising a framework to study the theoretical limits of predictability. The methods introduced in their framework are founded in information theory and have since been extensively applied in the area of mobility modeling and prediction. Later studies that built on [3] addressed either the specifics of the prediction problem (e.g., different formulations [4] of the individual's change of location, analyzed different contexts of mobility) or the shortcomings of the original approach (that relied on coarse spatio-temporal granularity). Authors in [5] used Wireless LAN (WLAN) traces from a university campus network and reported multi-modal entropy distributions which can be partially explained by the demographics of the population (*i.e.*, age, gender, major of studies). Other entropy based studies include vehicular mobility [6], [7], [8], online social behavior [9], [10], complex systems [11], cellular network traffic [12] and public transport utilization [13]. In addition, devices' form factor affects the mode of usage and varied traffic profiles ([14], [15], [16], [17]), but these studies either do not consider predictability or do not account for different spatio-temporal resolutions. We have chosen our methods based on the literature to measure and compare both theoretical and practical limits of predictability for *Flutes* and *Cellos*, with varying degrees of spatio-temporal granularity, while also looking at the correlation of prediction accuracy with mobility and network traffic profiles using extensive fine-granularity traces (based on our earlier work in [17]).

The *main* questions addressed in this study are: i. How different are *Flutes* and *Cellos* in terms of predictability? ii. How does the predictability of these device types change with different *spatio-temporal granularity* (5, 15, 30 min, 1 hour and 2 hours; access point and building level)? iii. Does the *choice of method* or predictor (*e.g.* Markov Chain, neural networks such as LSTM and CNN, BWT or LZ based estimators, which are introduced in Section II) significantly alter the answers to aforementioned questions?

This study provides the following main contributions: 1. Quantifying the differences of *Flutes* and *Cellos* for prediction analysis, evaluated on a real-world large-scale dataset. 2. Comparison of several well-known algorithms (Markov Chains, Neural Networks) and LZ/BWT-based theoretical bounds across different time and space scales for Flutes and Cellos. 3. Use of prediction accuracy as part of the user profile for modeling, and investigation of its correlation with a combination of network traffic and mobility features.

The paper is structured as follows: First, the main approach and methods are presented in Sec. II. Then, the details of the dataset and experiment setup are discussed in Sec. III. The experiment results are presented in Sec. IV. Sections V and VI present the discussion on potentials implications of the findings and conclude the paper.

II. Main Approach & Methods

We investigate two methods to measure predictability; a theoretical method based on entropy, and a systems method based on practical predictor algorithms. Following we provide the entropy estimation based definition and discuss the different algorithms studied in this paper, including a reference-point Markov Chains approach, and a more sophisticated deep learning approach.

A. Entropy Estimation

Entropy is defined as the level of order (or disorder) of a system, and is founded on information theory. It has been adopted in previous studies to establish bounds on predictability under certain assumptions [3], [4]. We utilize it in our study to gauge the performance of our practical predictors. For a random process, this metric is sensitive to both the relative frequency of events and their inter-dependencies [13]. To estimate a baseline of predictability, we compute the *time-uncorrelated* entropy ($\mathcal{S}^{\text{unc}}$) which only takes into account the frequency of the observed events. For the upper-bound of predictability we compute two *time-correlated* estimators based on compression algorithms ($\mathcal{S}^{\text{lz}}$ and $\mathcal{S}^{\text{bwt}}$) which also consider the memory of the system. We define *maximum predictability* as the probability of predicting the most likely state of x_i given a state x_j, which is computed from the entropy S of a given sequence of events based on [3], with the refinements proposed by [4]. For a complete description on *entropy estimation*, we kindly refer the reader to [18] and [19].

B. Predictors

Markov Chain-based predictor: A Markov chain (MC) with a discrete state space has been applied for user mobility prediction [20], [21]. In an order-k Markov predictor, the state space consists of tuples of k location names (e.g., AP), where the next location prediction depends solely on the most recent preceding k-tuple. We build the model on the data so that observed k-tuples comprise the states. The transition probabilities are learned based on the frequency of appearances of such a transition in observations. The probability for a transition from the current state $S = X_i X_{i+1} ... X_j$ to $X_{i+1} X_{i+2} ... X_j X_{j+1}$ where $j - i = k$ and each X_i is the symbol for each location, is represented as $P(X_{j+1} = c \mid S = X_i X_{i+1} ... X_j)$ for all c observed in data and is learned based on the reappearance frequency of such a sequence. If the predictor of order k encounters a new sequence that has never seen before, it falls back to the lower, $k-1$ order recursively. The base case is O(0) which is simply the frequency distribution of all symbols observed so far.

Deep learning: Recent approaches to sequence prediction use deep Recurrent Neural Networks (RNN) or Convolutional Neural Networks (CNN). Recurrent neural networks have loops within their cells, allowing information to persist and thus enabling the neural network to connect previous information to make a reasonable prediction of the future. Certain types of RNNs are capable of learning long-term dependencies. There are multiple variants of RNNs, including Long short-term memory (LSTM) [22] and Gated Recurrent Unit (GRU) [23]. These networks can learn dynamic temporal patterns and have successfully been applied in speech recognition, text-to-speech engines and predicting next location [24], [25].

CNNs learn convolutional filters to extract latent information across the data (i.e. 1D CNNs learn different temporal locality patterns) and use that information for predicting the next location. In our study, we use a multi-layer LSTM and 1D CNN to predict movements of users based on similar input tuples used for MC-based predictors. Neural networks are computationally expensive and require hyper-parameter tuning. Thus the deep model is run only on a sample of users in this study. One goal of this study is to analyze the payoff (and cost) of adding complexity to the predictor (e.g. LSTMs), versus the simpler MC-based predictors.

III. Datasets & Experimental Setup

To study the regularity of human behavior, we performed a data-driven analysis applying our methods to a university campus WiFi traces from the University of Florida. The datasets were collected from networks providing wireless access to a large number of portable devices via access points deployed in non-residential areas, including classrooms, computer laboratories, libraries, offices, administrative premises, cafeterias, and restaurants.

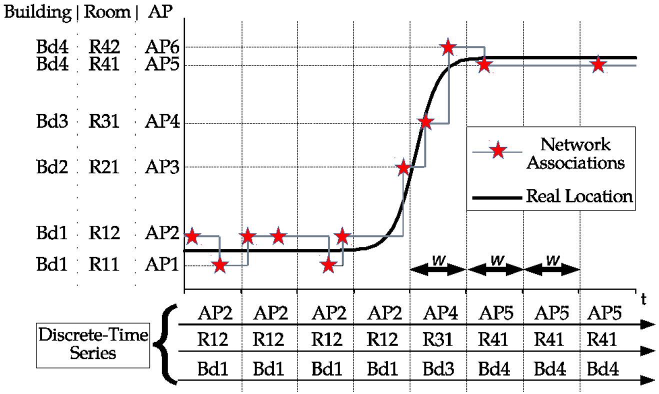
Every trace entry contains a unique user identifier (*uuid*), time-stamp and an access point unique identifier (*apid*). Based on the *apid*'s string we are able to identify the building as well as the room in which an access point (AP) was located. Only the geographical coordinates of buildings are known. Table II contains a brief summary of the dataset with mean (μ) and standard deviation (*std*), where N_{ap} is number of unique access points observed per device, N_{day} number of unique days with at least one record, N_{rec} number of records during data collection, and *total* number of devices available for at least 7 days and accessed more than 5 APs.[1]

A. UF traces

The UF traces were collected for 16 months (September/2011-December/2012) and contain over 1700 wireless access points (APs) deployed in 140 buildings which were used by 300K devices. A sample (sythentic) record is shown in Table I. Its raw records were captured from associations and sessions timeout in which the unique user id (*uuid*) was the MAC address. These *uuid* although hashed, still contained the Organizationally Unique Identifier (OUI)[2] allowing us to distinguish *Flutes* and *Cellos*, as detailed in [17]. All collected WiFi traces are processed as discrete time-series, defined next.

B. Discrete-time Series

Given a set a of timely ordered events $X = \{x_t : t = 1, \cdots, n\}$, where x_t is the realization of X at time t for $t \in T$, we say that a timeseries is *discrete* if T are measurements taken at successive times spaced at uniform intervals w, also referred to as sampling rate (defining the temporal granularity).



AP2	AP2	AP2	AP2	AP4	AP5	AP5	AP5
R12	R12	R12	R12	R31	R41	R41	R41
Bd1	Bd1	Bd1	Bd1	Bd3	Bd4	Bd4	Bd4

Fig. 1. Location of the device is sampled at a constant rate.

Figure 1 depicts an example of how the real location of a device is sensed by the wireless management system through AP associations (red stars) and finally how the discrete-time series is obtained. For a given sampling time window w, our *discrete-time* series may result in different sequences depending on whether we choose an AP or a building as the level of spatial resolution.

From Figure 1, for the first 4 time steps the device switched its associated AP without a real location change. This switch in AP association can be triggered by the mobile device (e.g. stronger wireless signal) or by the network management system (e.g. load balancing).

Note that it is important to define the resolution for *space* and *time*, *i.e.,* how big a location is in space (or point-of-interest) and how often we are going to sample from the input signal. In this example, larger values of w could eliminate this *ping-pong* effect of switching between APs without actually moving, but also cause loss of information when the user transits from one location to another. On the contrary, very small values of w could over-sample long periods when the user is not moving. Similarly, different values of spatial resolution could mitigate noise but eliminate information from the traces. Choosing these parameters is often influenced by the characteristics of the available dataset as well as the targeted application of the study.

Step Value: A weighing mechanism is used to pick the corresponding location to represent a time step. During a time interval, we weigh every observed location of the device with the duration of time at that location and pick the one with the highest weight to represent that step. We assign a user to a specific location ℓ in the time interval δt between an association at ℓ and the next association at any other location, but only if $\delta t < t_{\max}$. After $t_{\max}$ the device will be in an *unknown* state [3] until the next network event which will reveal its location for future steps.

C. Experiments

The design of our experiments is based on our study's questions: i. How different are *Flutes* and *Cellos* in terms of predictability? ii. How does the predictability of these device types change with different spatio-temporal granularity? iii. Does the choice of method or predictor significantly alter the answers to aforementioned questions? Thus, we evaluated a matrix, involving *combinations* of the following dimensions:

- Device Types: *Flutes* vs. *Cellos*.
- Temporal Resolutions: 5 min, 15 min, 30 min, 1 hour and 2 hours.
- Spatial Resolutions: Access Points, and Buildings.
- Methods: A. Well known sequence prediction algorithms from machine learning literature (Markov Chains, Neural Networks) B. Entropy-based Estimations of predictability upper-bounds.

The experiments were implemented in Python, the neural networks were implemented using Tensorflow [26] and Keras. Training is carried out in an *online* manner and the evaluation is through providing a sliding window of k observations to the predictor and testing the prediction correctness of the next symbol. The *fraction of correct next symbol predictions* is the prediction accuracy metric.

IV. Experiment Results

A. Spatio-Temporal Resolutions

To answer the first two questions of this study, particularly ”ii. How does the predictability of these device types change with different *spatio-temporal granularity*?”, Table III summarizes the median accuracy of an LSTM predictor for Flutes and Cellos with different spatial and temporal granularity.

[1] Transient devices are not counted to ensure the analysis is carried out on devices that are mobile and benefit from predictive systems the most, while stationary devices (e.g. plugged-in Cellos) and guests that never return to campus are ignored.

[2] http://standards.ieee.org/faqs/regauth.html#17

TABLE I
AP LOGS SAMPLE DATA COLUMNS

User IP	UUID	AP name	AP MAC	Lease begin time	Lease end time
10.130.90.3	00:11:22:00:00:00	b422r143-win-1	00:1d:e5:8f:1b:30	1333238737	1333238741

TABLE II
STATISTICS PER DEVICE AVAILABLE FOR AT LEAST 7 DAYS & ACCESSED MORE THAN 5 APS.

	N_{ap}		N_{day}		N_{rec}		Total Devices
	μ	std	μ	std	μ	std	
UF	127.3	142.3	63.5	59.2	1861	5121	138028

The choice of granularity is application-dependent, for example, to predict foot traffic at buildings and congestion planning based on density, building level analysis is more appropriate. Cellos show more predictable behavior overall, as the fraction of correct next symbol predictions is higher for Cellos across the board. At the AP level, with longer time bins, the accuracy for both Flutes and Cellos decreases. This observation is in line with previous findings [4]. At 15min time intervals, the difference between Flutes and Cellos is at its maximum and drops and remains stable for longer time intervals. At the building level, the accuracy follows a less regular pattern but both Flutes and Cellos are most predictable at 5min intervals (due to repeats of the same location in the sequence). Cellos' accuracy drops for 30min bins and goes back up again. On the other hand, Flutes are more predictable in 30min bins than 15min, 1h or 2h bins.

Looking across all temporal bins, Fig 2 presents the empirical cumulative distribution function (ECDF) of prediction accuracy at AP and building spatial granularity. The "sit-to-use" *Cellos* show significantly higher predictability at every percentile; this is reasonable given their lower mobility [17] and mode of usage. In fact, prediction accuracy is highly correlated with other mobility and network traffic features of mobile wireless users, we will take a brief look at these correlations in Section V and Fig 3.

TABLE III
MEDIAN ACCURACY OF LSTM (SEQUENCE LEN. 40) FOR *Flutes* VS *Cellos*, 5MIN-2H TEMPORAL AND AP/BLDG SPATIAL GRANULARITY.

	AP		Building	
	F	C	F	C
5 min	33.22	42.25	44	63.4
15 min	21.42	36.9	34.53	58.06
30 min	21.88	27.39	39.56	50.78
1 hour	19.67	24.33	32.62	52.03
2 hour	17.17	22.5	32.6	59.62

B. Comparison of Methods

To answer the third question of this study, "iii. Does the choice of method or predictor significantly alter the answers to aforementioned questions?", here we compare the experiment results for different methods: 1) *MC*: Markov Chain 2) *LSTM*: A type of recurrent neural network 3) *CNN*: 1D Convolutional Neural Network 4) *Hr_LZ*: Theoretical predictability based on the Lempel-Ziv (LZ) entropy estimator 5) *Hr_BWT*: Theoretical predictability based on the Burrows-Wheeler transform (BWT) entropy estimator. A summary of comparisons is presented in Table IV, for temporal granularity of 1h and 15min, highlighting the difference of *Cellos* - *Flutes*.

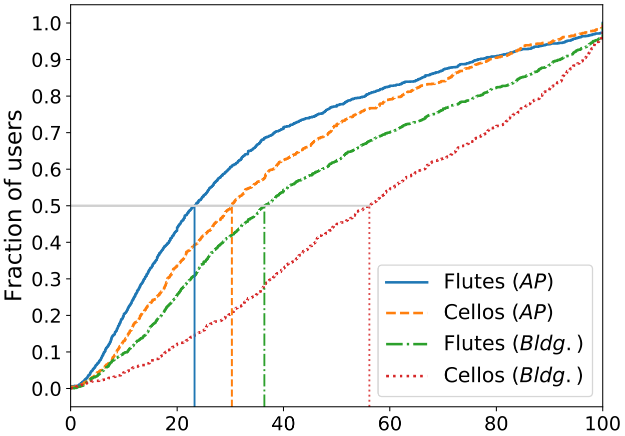


Fig. 2. ECDF of LSTM Prediction Accuracy for *Flutes* & *Cellos* at AP and Building spatial levels (all temporal levels combined, vertical lines denote medians, sequence length 40).

In all cases *Cellos* are more predictable than *Flutes*, regardless of the choice of method (with a minor exception of LZ predictor at 15min time and building level which might be due to intrinsic instability of LZ based estimator). The difference in median accuracy for *Flutes* vs *Cellos* is up to 25% (Building level, 15min window, sequence length 40, *Flutes* 33.97% vs *Cellos* 59.03%). Other temporal choices result in a similar pattern. Another notable observation is that while the neural networks are more complex, and require vastly more computing power, they only achieve modest increase compared to Markov Chains in *some* scenarios (e.g., *Cellos*, at the Bldg. level and Seq. Len. 40, from 48.56% to 52.5%). This is a trade-off that needs to be considered in the design of predictive caching systems. In addition, increasing the sequence length k (i.e. the number of previous time steps available to the predictor) impacts the Markov Chain model more than the neural networks. This is particularly pronounced for 15min time window, in fact, the neural networks do not lose much accuracy from increasing sequence length 5 to 40 in case of the 1h time window. Also, the theoretical LZ and BWT based estimators, show higher upper bounds compared with the best of the algorithms, with Seq. Len. 5 Markov Chains and CNNs being the closest practical algorithms for the

TABLE IV

Summary of Median Accuracy for *Flutes* vs *Cellos* with different methods (Diff is *Cellos* − *Flutes*) and sequence lengths for 15min and 1h time windows.

		AP, 1h			Bldg., 1h			AP, 15min			Bldg., 15min		
Seq Len	Predictor	F	C	Diff	F	C	Diff	F	C	Diff	F	C	Diff
5	MC	21.05	25.95	+4.90	38.25	53.50	+15.25	61.72	70.30	+8.58	75.00	87.60	+12.60
	LSTM	21.62	25.00	+3.38	35.03	50.00	+14.97	40.00	44.56	+4.56	52.44	65.56	+13.12
	CNN	16.45	24.27	+7.82	34.94	50.00	+15.06	50.00	59.80	+9.80	64.60	76.94	+12.34
10	MC	17.98	25.6	+7.62	36.72	50.28	+13.56	52.25	61.97	+9.72	68.00	82.25	+14.25
	LSTM	20.83	26.31	+5.48	37.50	50.66	+13.16	31.14	44.62	+13.48	45.38	64.56	+19.18
	CNN	18.06	22.62	+4.56	36.20	52.03	+15.83	49.20	58.80	+9.60	64.56	74.00	+9.44
20	MC	18.1	24.52	+6.42	36.28	49.94	+13.66	38.50	48.22	+9.72	57.30	74.94	+17.64
	LSTM	21.22	24.19	+2.97	36.12	50.78	+14.66	29.17	41.00	+11.83	43.62	61.47	+17.85
	CNN	18.44	23.60	+5.16	35.28	50.00	+14.72	37.84	48.12	+10.28	50.00	65.00	+15.00
40	MC	17.88	23.61	+5.73	35.1	48.56	+13.46	27.97	31,00	+3.03	47.12	65.80	+18.68
	LSTM	19.67	24.33	+4.66	32.62	52.03	+19.41	23.30	39.40	+16.10	33.97	59.03	+25.06
	CNN	18.75	23.97	+5.22	35.25	52.50	+17.25	27.62	44.70	+17.08	41.25	62.10	+20.85
	LZ	46.90	52.60	+5.70	58.78	66.40	+7.62	72.70	76.06	+3.36	79.60	79.10	-0.50
	BWT	66.44	69.44	+3.00	73.70	79.90	+6.20	83.30	88.06	+4.76	88.60	92.20	+3.60

15min case. The predictors are far behind in the 1h case, suggesting room for improvement via tuning for specific time and space granularities. The run time of LSTM is the longest, followed by CNN (not shown for brevity).

V. Discussion & Future Work

In this paper, we define our research problem as predicting the next symbol in a discrete-time series for users with two categories of devices. The accuracy is evaluated as the fraction of the next symbols predicted correctly.

While some earlier studies investigated a similar problem setup, our study has notable implications. For example, across device types, predictability can vary significantly. Also, with larger time windows such as 1 hour, it is easy to miss short stays (since one location visit with a duration of 31 minutes would result in other locations in that 1h window being ignored). On the other hand, a short time window results in multiple repetitions of the same location in the sequence, potentially achieving high prediction accuracy even when the method is not predicting the *transitions* well.

It is important to consider the device type, context, and application in order to choose an appropriate time and space granularity; the best performing method differs across these dimensions. Besides, the measured accuracy only considers an exact match to be correct, so even if the method predicts a nearby location to the actual location, it would count as incorrect. We plan to investigate measuring how far a predicted location is from the actual location and embed that information in the loss function of our neural networks for possible improvements in prediction.

Correlations with Mobility and Network Traffic: Figure 3 shows the correlation of prediction accuracy with a sample of features that describe the mobility or network traffic of users. PDT(W/E) and TJ(W/E) are mobility features while AAT(W/E) and AI(W/E) are traffic features. PDTW is the time spent at the user's preferred building (most common) on weekdays (PDTE for weekends). TJW is the total sum of jumps (distance) for the weekdays while TJE describes the same feature for weekends. AATW is the average of active time (as indicated by network usage) of the user for weekdays (AATE for weekends). AIW stands for average inter-arrival time of flows on weekdays, and AIE for weekends ([17], [27]).

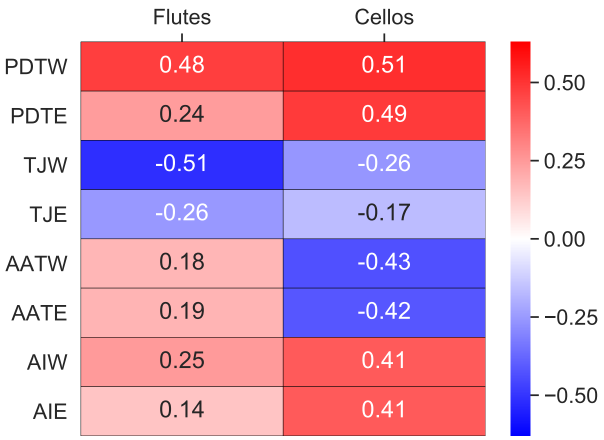


Fig. 3. Pearson Correlation of Prediction Accuracy with several Mobility and Network Traffic Features.

The results present significant correlations between the prediction accuracy, with not only the *mobility features*, but also *network traffic features*. These correlations vary across device types (*Flutes* vs *Cellos*), and in time (*Weekdays* vs *Weekends*). This is a very important observation for the design of *predictive caching* systems, importantly, it might be possible to improve prediction of *where* the user is going based on network traffic profile while noting the different modes of usage based on device types. We leave the investigation of such improvements to future work.

Integrated Mobility-Traffic Modeling: Given the observed correlations, we hypothesize that use of *predictability* as a feature in an integrated mobility-traffic generative model could lead to more realistic synthetic traces. Such a data-driven generative model would be an essential tool for network simulations and capacity planning. Notably, it can also be made *privacy preserving*, since collected traces would be replaced with realistic synthetic data that captures mobility, network traffic, predictability, and their relationships. Further study is beyond the scope of this work and is left for the future.

VI. Conclusion

In this work, we sought to answer three questions: i. How different are *Flutes* and *Cellos* in terms of predictability? ii. How does the predictability of these device types change with different *spatio-temporal granularity*? iii. Does the *choice of method* or predictor significantly alter the answers to aforementioned questions? For this purpose, we processed a large-scale dataset from a campus environment, and grouped the devices into two categories; and we chose a set of methods to make the comparisons, including Entropy-based estimators and popular algorithms such as Markov Chains and Neural Networks.

The results of experiments show the movements of *Cellos* (”sit-to-use”) are significantly more predictable than *Flutes* (up to 25% difference in accuracy). This pattern is consistent across various temporal granularities (5 min to 2 hours), spatial granularities (Access Point and Building level), and for different methods (Markov Chains, Neural Networks, Entropy-based Estimators). We illustrate that the performance of predictors depends strongly on the span of temporal bins. Markov Chains tend to outperform deep learning models in shorter time-bins while LSTMs and CNNs usually show a higher accuracy in longer time-bins. CNNs have mostly similar accuracy to LSTMs in the latter case but have significantly better run time on a modern GPU. We also found significant correlations among prediction accuracy, *mobility features*, and also *network traffic features*, an important observation for the design of *predictive caching* systems where it might be possible to improve mobility prediction based on network traffic profile. We plan to further investigate the use of *predictability as a feature* in an integrated mobility-traffic generative model, and its application in state-of-the-art predictive caching systems.

Acknowledgement

This work was partially funded by NSF 1320694. We gratefully acknowledge the support of NVIDIA Corp. with the donation of the Titan Xp GPU used for this research.

References

[1] V. Siris, X. Vasilakos, and D. Dimopoulos, “Exploiting mobility prediction for mobility, popularity caching and dash adaptation,” in *WoWMoM*, 2016.

[2] N. Lathia, “The anatomy of mobile location-based recommender systems,” in *Recommender Systems Handbook*. Springer, 2015.

[3] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, “Limits of predictability in human mobility,” *Science*, 2010.

[4] G. Smith, R. Wieser, J. Goulding, and D. Barrack, “A refined limit on the predictability of human mobility,” *PerCom*, 2014.

[5] P. Cao, G. Li, A. Champion, D. Xuan, S. Romig, and W. Zhao, “On human mobility predictability via WLAN logs,” in *Proc. INFOCOM*, Apr. 2017.

[6] Y. Li, D. Jin, P. Hui, Z. Wang, and S. Chen, “Limits of predictability for large-scale urban vehicular mobility,” *IEEE Transactions on Intelligent Transportation Systems*, 2014.

[7] J. Wang, Y. Mao, J. Li, Z. Xiong, and W. X. Wang, “Predictability of road traffic and congestion in urban areas,” *PLoS ONE*, 2015.

[8] R. Gallotti, A. Bazzani, M. D. Esposti, and S. Rambaldi, “Entropic measures of individual mobility patterns,” *Journal of Statistical Mechanics: Theory and Experiment*, no. 10, 2013.

[9] T. Takaguchi, M. Nakamura, N. Sato, K. Yano, and N. Masuda, “Predictability of conversation partners,” *Physical Review X*, 2011.

[10] R. Sinatra and M. Szell, “Entropy and the predictability of online life,” *Entropy*, vol. 16, no. 1, pp. 543–556, 2014.

[11] R. Hanel and S. Thurner, “A comprehensive classification of complex statistical systems and an axiomatic derivation of their entropy and distribution functions,” *Epl*, vol. 93, no. 2, 2011.

[12] X. Zhou, Z. Zhao, R. Li, Y. Zhou, and H. Zhang, “The predictability of cellular networks traffic,” in *ISCIT 2012*, 2012.

[13] G. Goulet-Langlois, H. N. Koutsopoulos, Z. Zhao, and J. Zhao, “Measuring regularity of individual travel patterns,” *IEEE Transactions on Intelligent Transportation Systems*, 2017.

[14] G. Maier, F. Schneider, and A. Feldmann, “A first look at mobile hand-held device traffic,” in *PAM*. Springer, 2010.

[15] X. Chen, R. Jin, K. Suh, B. Wang, and W. Wei, “Network performance of smart mobile handhelds in a university campus wifi network,” *ACM IMC*, 2012.

[16] U. Kumar, J. Kim, and A. Helmy, “Changing patterns of mobile network (WLAN) usage: Smart-phones vs. laptops,” *IWCMC*, 2013.

[17] B. Alipour, L. Tonetto, A. Yi Ding, R. Ketabi, J. Ott, and A. Helmy, “Flutes vs. cellos: Analyzing mobility-traffic correlations in large wlan traces,” in *IEEE INFOCOM*, 2018.

[18] Y. Gao, I. Kontoyiannis, and E. Bienenstock, “Estimating the entropy of binary time series: Methodology, some theory and a simulation study,” *Entropy*, vol. 10, no. 2, pp. 71–99, 2008.

[19] H. Cai, S. R. Kulkarni, and S. Verdú, “Universal entropy estimation via block sorting,” pp. 1551–1561, 2004.

[20] L. Song, D. Kotz, R. Jain, and X. He, “Evaluating location predictors with extensive Wi-Fi mobility data,” in *INFOCOM*, 2004.

[21] X. Lu, E. Wetter, N. Bharti, A. J. Tatem, and L. Bengtsson, “Approaching the limit of predictability in human mobility,” *Scientific reports*, vol. 3, 2013.

[22] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.

[24] J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural networks*, vol. 61, pp. 85–117, 2015.

[25] A. Karatzoglou, A. Jablonski, and M. Beigl, “A seq2seq learning approach for modeling semantic trajectories and predicting the next location,” in *ACM SIGSPATIAL*, 2018.

[26] “TensorFlow: Large-scale machine learning on heterogeneous systems,” software available from tensorflow.org.

[27] B. Alipour, M. Al Qathrady, and A. Helmy, “Learning the relation between mobile encounters and web traffic patterns: A data-driven study,” in *ACM MSWIM*, 2018.