Beyond The Limits of Predictability in Human Mobility Prediction: Context-Transition Predictability

Chao Zhang, Kai Zhao, and Meng Chen *Member, IEEE*

Abstract—Urban human mobility prediction is forecasting how people move in cities. It is crucial for many smart city applications including route optimization, preparing for dramatic shifts in modes of transportation, or mitigating the epidemic spread of viruses such as COVID-19. Previous research propose the maximum predictability to derive the theoretical limits of accuracy that any predictive algorithm could achieve on predicting urban human mobility. However, existing maximum predictability only considers the sequential patterns of human movements and neglects the contextual information such as the time or the types of places that people visit, which plays an important role in predicting one's next location. In this paper, we propose new theoretical limits of predictability, namely Context-Transition Predictability, which not only captures the sequential patterns of human mobility, but also considers the contextual information of human behavior. We compare our Context-Transition Predictability with other kinds of predictability and find that it is larger than these existing ones. We also show that our proposed Context-Transition Predictability provides us a better guidance on which predictive algorithm to be used for forecasting the next location when considering the contextual information. Source code is at https://github.com/zcfinal/ContextTransitionPredictability.

Index Terms—limits of predictability; entropy; location prediction; predictive algorithm; human mobility

1 Introduction

Cities are the loci of economic activity and innovation. Today, about 50% of the world's population lives in cities, and this number will rise to 68% by 2050[1]; North America is already 80% in cities, and the number will be 90% by 2050. At the same time, many cities face challenges around transportation, resource consumption, housing affordability, and aging infrastructure. Data, along with big data analytics, can help significantly deal with these challenges: our ability to collect, transmit, and store data, creates an opportunity to leverage and analyze these data to make cities more productive, livable, equitable, and resilient.

Urban human mobility prediction, which is forecasting how people move in cities, can help with many challenges faced by cities, including route optimization [17], preparing for dramatic shifts in modes of transportation [37], or mitigating epidemic spread of viruses such as COVID-19 [12]. Taking COVID-19 as an example; by analyzing anonymized and aggregated mobile data, it has been found that human mobility alone can explain up to 92% of the initial spread of COVID-19 in cities [32]. Research study [15] shows that early quarantine of infected individuals and their contacts could significantly reduce the number of COVID-19 infections by 78.2-99.3%. Understanding and predicting how people move in cities can help to make decisions on quarantining infected individuals, their family members, and their contacts, workplace social distancing, and school closure [16].

One important research question in human mobility prediction is on how predictable they are. Song et al. [28] analyze human mobility and firstly propose this concept (the limits of predictability) in their paper published in *Science* in 2010, which is defined based on entropy while considering the randomness and regularity of human movements. Human movements have strong regularity; people tend to go to working place from home, and go shopping from working place. Such human behavior can be utilized for defining the maximum predictability of one person's trajectory. The limits of predictability have been widely used in human mobility modeling and prediction in data mining area in the past ten years [5], [20], [23], [29].

With the development of recent technologies, especially with the advance of deep learning models, we observe that many predictors have reached prediction accuracy beyond the limits of predictability [38]. The problem with previous defined limits of predictability [28], is that it mainly captures the sequential patterns of human mobility data without considering the contextual information. Contextual information such as the time or the types of locations that people visit, plays an important role in predicting one's next location. For example, people would like to go to work in the morning during weekdays and visit a shop in the evening after coming back home. Thus the person has a much higher chance of visiting her/his office in the morning, and going to a shopping mall in the evening from home (see illustration

- *Chao Zhang and Meng Chen are with the School of Software, Shandong University, Jinan, China. Email: 201800301086@mail.sdu.edu.cn; mchen@sdu.edu.cn.*
- *Kai Zhao is with Robinson College of Business, Georgia State University, Georgia, USA. Email: kzhao4@gsu.edu*
- *Chao Zhang and Kai Zhao are co-first authors and contribute equally to this paper.*
- *Corresponding author: Meng Chen.*

[1] 68% of the world population projected to live in urban areas by 2050, says UN. https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html

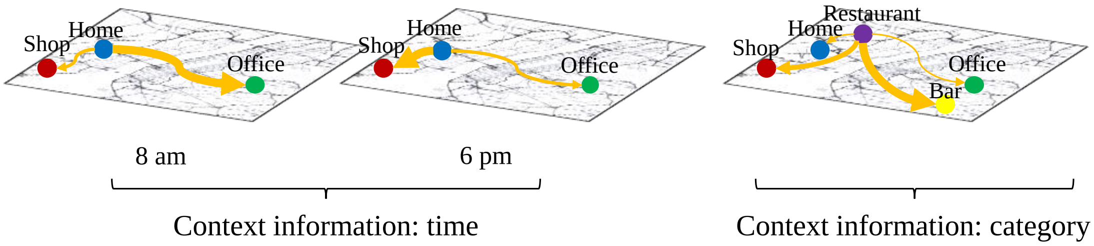


Fig. 1: Illustration of human mobility given the context information

in Fig. 1) if we consider the temporal information. Similarly, if a person is at a restaurant, she/he will have a higher chance of going to a bar after this current location, instead of going to other places. A predictor which captures the context information such as the time, the weather and the social events, will have higher prediction accuracy.

To address the issue that previously defined limits of predictability do not consider the contextual information, we propose **Context-Transition Predictability** as the limits of predictability. Instead of the real entropy used in the previous research [20], [38], we define the conditional transition entropy to derive the limits of predictability, which captures both the sequential orders of human mobility and the context information it contains. We show that our newly defined limits of predictability are actually larger than the previous maximum predictability, enlarging the limit bound.

In addition to that, we reevaluate most commonly used location predictive algorithms, and examine which algorithm could reach such limits. We observe that the embedding predictive algorithm outperforms all other predictors, and is able to approach the limits of predictability. We also observe that there is a high correlation between our proposed Context-Transition Predictability and prediction accuracy of the predictive algorithm that considers the contextual information when training (e.g., the embedding predictor in our case).

Our experimental results reveal that, as a measurement for both the sequential regularity and the contextual information, the proposed Context-Transition Predictability can capture the theoretical limits of human mobility prediction and provide an upper bound of predictive power for mobility data. Our proposed Context-Transition Predictability provides an important strategy to guide the design and the selection of human mobility predictive algorithms. The prediction of human mobility is heavily dependent on a user's contextual information such as the user type (tourists vs local persons), the time (weekdays vs weekends), and the user's current location type (home vs restaurants). The embedding predictor that includes all such contextual information when training is able to provide the best prediction accuracy and approaches the limits of predictability under the guidance of Context-Transition Predictability.

Summary of Contribution: In this study, we make the following contributions.

- First, methodologically, we are among the first to consider both human movements and their contextual information to enlarge the limit bound of predictive algorithms and define the novel limits of predictability, named Context-Transition Predictability.
- Second, empirically, we show that our proposed Context-Transition Predictability provides us with guidance on which predictors to be used for forecasting the urban human mobility. We observe that there is a high correlation between our proposed Context-Transition Predictability and prediction accuracy of the machine learning algorithms such as the embedding predictor. The prediction of a user's next location is heavily dependent on the contextual information and a predictive algorithm that captures such contextual information can provide better prediction accuracy.
- Third, substantively, we empirically show that incorporating the knowledge of contextual information of human mobility can indeed improve the limits of predictability for location prediction. We compare Context-Transition Predictability with other kinds of predictability and find that it is larger than these existing predictability. Human mobility exhibits strong regular patterns when considering the contextual information and we can utilize such information to better predict human mobility.

2 Related Work

2.1 Human Mobility Predictability

Human mobility exhibits sequential regularity: people tend to go to work from home, and go shopping after work. Gonzales et al. [9] investigate the traces of over ten-thousands mobile phone users and observe that human mobility exhibits strong spatio-temporal regularity. Specifically, they find that users are more likely to visit a few frequently visited locations such as home or working places. Such spatio-temporal regularity can be used to determine the predictability of human mobility. Song et al. [28] propose the limits of predictability based on entropy to measure a user's spatio-temporal mobility pattern. They observe that the prediction accuracy of a predictive algorithm could be up to 93% if it captures all spatio-temporal regularities. Lian et al. [20] study the limits of location predictability using a check-in social network data set. De Domenico et al. [7] investigate the predictability of human mobility by considering the movement of people with correlated mobility patterns. Teixeira et al. [29] propose to explain the variability in the predictability of human movements using two measures named regularity and stationarity.

Recently, researchers examine the relations between human mobility predictability and performances of predictive algorithms. Lu et al. [23] take the Markov model as the predictive algorithm and investigate the relations between the predictability and the prediction accuracy, and find that there exists a positive correlation between them. Zhao et al. [38] find that when human mobility exhibits a strong predictability, a Markov predictive algorithm with half seconds training time easily outperforms a deep learning algorithm with over six hours training time.

Overall, the above studies suggest that there are limits of predictability of human mobility prediction based on the sequential regularity of one's movements. However, the existing research studies neglect the contextual information of a person's movements, which plays an important role in predicting their locations. Our work advances prior studies by proposing novel limits of predictability, which not only captures the sequential regularity of human mobility, but also considers the key contextual information such as time and the property of locations. We empirically show that our proposed Context-Transition Predictability enlarges the limit bound for human mobility prediction.

2.2 Next Location Predictor

The next location prediction problem has becoming more and more interesting with the rapid growth of trajectory data (e.g., vehicle trips or user check-ins). Most traditional methods adopt Markov models or frequent pattern mining methods to capture a person's sequential behavior and to make location prediction given the recently visited locations. For example, Chen et al. [2] propose multiple variable-order Markov models which capture individual, collective, and temporal patterns to predict the successive locations. Qiao et al. [26] propose PrefixTP, a Prefix-projection-based Trajectory Prediction algorithm, which predicts the next locations based on the frequent trajectory patterns.

Recently, some methods adopt the matrix/tensor factorization technique to study the problem of next location prediction. Lian et al. [19] present a collaborative filtering based method which encodes users' check-in patterns in a latent space to make successive point-of-interest recommendation. Zhao et al. [42] propose a spatial-temporal latent ranking method called STELLAR to model the interactions among user, location, and time for location prediction. Chen et al. [4] employ the matrix factorization method to model both the global and local context of users' trajectories to generate representations of trajectory attributes, which could be used in many tasks including location prediction. Feng et al. [8] propose a personalized ranking embedding method for check-in location recommendation and prediction, which integrates individual behavior preference, mobility sequential information, and geographical contextual influence.

Deep learning methods are also widely employed in next location prediction as well, as they have gained a breakthrough in mobility data mining recently. Luca et al. [24] survey recent studies and propose a perspective on the leading deep learning solutions (including fully connected networks, recurrent neural networks, attention mechanisms, and convolutional neural networks) to next location prediction. In particular, some methods focus on mining the sequential patterns from human movement data. For instance, Xiao and Nian [31] propose a hybrid LSTM method for vehicle location prediction. They convert a vehicle trajectory into the input of LSTM using a transformation method, and then predict the successive locations via the proposed model. Zhao et al. [40] propose a stacked recurrent neural network framework to discover users' sequential patterns with their check-in sequences.

Further, some studies aim at capturing both the sequential information and the contextual information simultaneously for location prediction. For example, De Brébisson et al. [6] model GPS point sequences and context information (e.g., driver identity and departure time) based on fully connected networks to predict the destination of a trajectory. Liu et al. [22] consider the temporal and spatial influence of human mobility behavior and present a Spatial Temporal Recurrent Neural Network for location prediction. Kong and Wu [13] propose a hierarchical LSTM model to capture the spatial-temporal factors. Liu et al. [21] propose Context-Aware Recurrent Neural Networks which model both transition contexts (i.e., time intervals between adjacent behaviors) and complex contexts including time, location, and weather to address the problem of sequential recommendation. Lv et al. [25] treat trajectories as images and leverage CNNs to extract spatial features and combine them with other contexts such as starting time and personal information to make location prediction. All these works above have shown the significance of combining contextual and sequential information for better location prediction.

In addition, some works employ the embedding methods for location prediction. For example, Chen et al. [3] propose an embedding method to embed the time, current and next locations to predict future visiting places. Chang et al. [1] propose a text-aware location embedding algorithm, which utilizes the content of a location to improve the performance of next location recommendation. Ying et al. [36] consider two properties (namely the asymmetric property of transitions between consecutive locations in check-in sequences, and dynamic user preference at different time) of check-ins and propose a time-aware metric embedding approach with asymmetric projection for next location recommendation. Zhao et al. [41] propose a neural network framework which jointly trains the location context prediction and the next location recommendation.

We observe that there are a lot of deep learning algorithms proposed recently for human mobility prediction. Different from these existing methods, we do not try to propose a new prediction model in our study. Instead, we propose novel limits of predictability which considers both sequential and contextual information, to provide a theoretical upper bound of prediction accuracy for all these deep learning methods. In addition, we empirically show that our proposed Context-Transition Predictability also provides an important strategy to guide the design and the selection of all existing human mobility deep learning methods.

3 Problem Definition

3.1 Concepts

Definition 1 (Location). *A **location** l refers to a point or a region where the position of a user u is recorded.*

TABLE 1: Notations and descriptions.

Notations	Descriptions
u, l, t, tr	user, location, time, transition
T	trajectory
$\mathcal{L}$	set of locations
$\mathcal{E}_u^l$	set of location sequences of user u
$\mathcal{E}_u^t$	set of time sequences of user u
$E^{Shannon}$	Shannon entropy
E^{Real}	real entropy
$E_{Transition}^{Context}$	Context-Transition entropy
Π	predictability of an arbitrary predictive algorithm $\mathcal{A}$
Π^{max}	upper bound of predictability Π,$\Pi \leq \Pi^{max}$

Definition 2 (Trajectory). *A* **trajectory** *T_n is a time-ordered sequence of locations: $\langle(l_1, t_1), (l_2, t_2), \dots, (l_n, t_n)\rangle$ visited by a user u, where n is the number of visited locations and t_n is the time that u visits l_n.*

Definition 3 (Location Sequence). *Given a trajectory $T_n = \langle(l_1, t_1), \dots, (l_n, t_n)\rangle$, its* **location sequence** *is defined as $\langle l_1, l_2, \dots, l_n\rangle$.*

Definition 4 (Transition). *Given a sequence $\langle l_1, l_2, \dots, l_n\rangle$, $l_i \rightarrow l_{i+1}$ is defined as a* ***transition*** *tr if a user could reach l_{i+1} from l_i directly without visiting any other location.*

3.2 Predictability in Human Mobility

Given that a user has a trajectory T_i and is currently at location l_i, we assume that there is a prediction algorithm $\mathcal{A}$, which could estimate a probability distribution $P(l_{i+1} = l|T_i)$ for the next visited location. We choose location l_{pred} with the maximum probability as the predicted one:

$$l_{pred} = \arg\max_{l \in \mathcal{L}} P(l_{i+1} = l|T_i), \tag{1}$$

where $\mathcal{L}$ is the set of unique locations. Intuitively, if $P(l_{i+1}|T_i)$ reaches a significant peak at location l_{pred}, the next location could be predicted accurately with high confidence; if $P(l_{i+1}|T_i)$ has a flat distribution, the next location has a small probability to be predicted accurately.

Formally, we aim to solve the following problem:

Problem 1. *Given a trajectory T_i and any predictive algorithm $\mathcal{A}$, considering both the sequential patterns and the contextual information (e.g., visiting time of locations) of trajectories, find the maximum predictability Π^{max} that $\mathcal{A}$ can reach when trying to guess the next location l_{i+1}.*

Note that the maximum predictability Π^{max} is an expression of human behavior, which relies on both the trajectory data and the prediction tasks. In this work, we choose next location prediction as the prediction task. Specifically, given a trajectory T_i of user u, we predict the first location that will be visited by u after l_i.

4 Estimating Predictability of Human Mobility

We first investigate the predictability estimate based on entropy with respect to the location sequences, and then introduce the proposed Context-Transition Predictability that models the contextual information of trajectories.

4.1 Entropy of Location Sequences

We first explore the predictability that only considers the location sequences of trajectories, i.e., deriving the predictability that any prediction algorithm $\mathcal{A}$ can achieve when trying to predict l_{i+1} by calculating $P(l_{i+1} = l|l_1, \cdots, l_i)$. As a user's location sequences could be regarded as a stochastic process, we propose to leverage the entropy rate of the stochastic process to compute the predictability of location sequences. Here we first introduce two existing measures of entropy: the Shannon entropy $E^{Shannon}$ and the real entropy E^{Real}.

Shannon Entropy. Given the set $\mathcal{E}_u^l$ of location sequences of user u, the Shannon entropy $E^{Shannon}(\mathcal{E}_u^l)$ is defined as

$$E^{Shannon}(\mathcal{E}_u^l) = -\sum_{i=1}^{N^{(u)}} P(l_i)\log_2 P(l_i), \tag{2}$$

where $N^{(u)}$ is the number of locations that u visits in $\mathcal{E}_u^l$, and $P(l_i)$ is the frequency that location l_i occurs in $\mathcal{E}_u^l$. For example, given two location sequences of a user: $\langle l_1, l_2, l_3\rangle$ and $\langle l_2, l_1, l_3, l_4\rangle$, we can get $E^{Shannon} = -\frac{2}{7}\log_2\frac{2}{7} - \frac{2}{7}\log_2\frac{2}{7} - \frac{2}{7}\log_2\frac{2}{7} - \frac{1}{7}\log_2\frac{1}{7}$. Note, the Shannon entropy does not take the order of locations in the trajectories into account.

Real Entropy. Given the set $\mathcal{E}_u^l$ of location sequences of user u, the real entropy $E^{Real}(\mathcal{E}_u^l)$ is defined as

$$E^{Real}(\mathcal{E}_u^l) = -\sum_{s \subset \mathcal{E}_u^l} P(s)\log_2 P(s), \tag{3}$$

where $P(s)$ is the probability of finding a particular sequence s in $\mathcal{E}_u^l$. Note, the real entropy models the order of locations in the users' trajectories, which is different from the Shannon entropy.

The problem of finding all the sequences from $\mathcal{E}_u^l$ has exponential computational complexity. Therefore, it is impractical to compute it directly. Here we adopt the Lempel-Ziv estimator [14], which is an approximation method and is able to converge to the value of the real entropy quickly. As such, the real entropy E^{Real} of a location sequence $\langle l_1, l_2, \dots, l_n\rangle$ is calculated by

$$E^{Real} \approx \frac{\ln n}{\frac{1}{n}\sum_{i=1}^{n}\Lambda_i}, \tag{4}$$

where Λ_i represents the length of the shortest subsequence starting at position i which does not previously appear from position 1 to $i-1$ in the location sequence.

4.2 Context-Transition Entropy of Trajectories

Both the Shannon entropy and the real entropy do not consider the contextual information (e.g., visiting time of locations, semantic categories of locations) in the predictability of human mobility. Considering the characteristics of trajectories, we propose a new method to calculate the entropy of trajectories, namely **Context-Transition Entropy**, which is inspired from both the Shannon entropy and the real entropy, modeling both the context information and the transition patterns of human mobility.

As we know, human mobility patterns vary with contextual information. One such example is the temporal information, e.g., a user is most likely to visit restaurants around

12:00 p.m., and visit a bar around 22:00 p.m. Therefore, in addition to the location sequences, we also need to consider the contextual information that a user visits every location in the trajectories when estimating the predictability. We explore whether lower entropy values are obtained when the contextual information is utilized. Other information (e.g., categories of locations) could also be utilized as the context, which is investigated in the experiments in Section 8.3. Note that at least one of the contextual information must be present if we want to define the context entropy, no matter the temporal information, the location category information or other kinds of contextual information.

To define the Context-Transition entropy, we first fetch the location sequence $\mathcal{E}_u^l = \langle l_1, l_2, \ldots, l_i \rangle$ and the corresponding contextual information sequence $\mathcal{E}_u^t = \langle t_1, t_2, \ldots, t_i \rangle$ from a user u's trajectory sequence $T_i = \langle (l_1, t_1), (l_2, t_2), \ldots, (l_i, t_i) \rangle$. Then we compute the conditional entropy $E(\mathcal{E}_u^l | \mathcal{E}_u^t)$ as follows:

$$E(\mathcal{E}_u^l | \mathcal{E}_u^t) = E(\mathcal{E}_u^l) - I(\mathcal{E}_u^l, \mathcal{E}_u^t), \tag{5}$$

where $E(\mathcal{E}_u^l)$ is the entropy value of these location sequences $\mathcal{E}_u^l$, $I(\mathcal{E}_u^l, \mathcal{E}_u^t)$ is the mutual information between $\mathcal{E}_u^l$ and $\mathcal{E}_u^t$, and $E(\mathcal{E}_u^l | \mathcal{E}_u^t)$ measures how much knowing $\mathcal{E}_u^t$ could assist in calculating the entropy value of $\mathcal{E}_u^l$.

It has been shown that a user's next visited location is heavily affected by the just passed locations [3]. Therefore, we need to capture the order of locations in a trajectory to compute $E(\mathcal{E}_u^l)$. We compute the Shannon entropy for the location transitions instead of these individual locations in the trajectories. The entropy $E(\mathcal{E}_u^l)$ is defined as

$$E(\mathcal{E}_u^l) = - \sum_{i=1}^{N^{(tr)}} P(tr_i) \log_2 P(tr_i), \tag{6}$$

where $N^{(tr)}$ represents the number of transitions in $\mathcal{E}_u^l$, and $P(tr_i)$ denotes the frequency that the transition tr_i occurs in $\mathcal{E}_u^l$. From the perspective of location transitions and the contextual information associated with each transition, we compute the mutual information $I(\mathcal{E}_u^l, \mathcal{E}_u^t)$ as

$$I(\mathcal{E}_u^l, \mathcal{E}_u^t) = \sum_{tr \subset \mathcal{E}_u^l} \sum_{t \in \mathcal{E}_u^t} P(tr, t) \log \frac{P(tr, t)}{P(tr) P(t)}, \tag{7}$$

where the probability distributions $P(tr, t)$, $P(tr)$ and $P(t)$ could be computed based on the training data.

We compute the location transition entropy in Equation (6) based on the Shannon entropy, instead of the real entropy. The reason is that the location subsequence obtained by the Lempel-Ziv estimator (which is an approximation method (see Equation (4)) for computing the real entropy) has a varying length and thus it is difficult to match it with the corresponding contextual information to calculate the mutual information. Therefore, we calculate the location transition entropy based on the Shannon entropy, which could capture the sequential patterns of trajectories and the mutual information between location sequences and the corresponding context sequences easily. Further, we also consider subsequences with different orders to capture the sequential patterns of trajectories, which will be investigated in the experiments in Section 6.3.

Combining Equations (5), (6) and (7), we compute the Context-Transition entropy for the trajectories of each user:

$$\begin{aligned} E_{Transition}^{Context}(\mathcal{E}_u^l | \mathcal{E}_u^t) = & - \sum_{i=1}^{N^{(tr)}} P(tr_i) \log_2 P(tr_i) - \\ & \sum_{tr \subset \mathcal{E}_u^l} \sum_{t \in \mathcal{E}_u^t} P(tr, t) \log \frac{P(tr, t)}{P(tr) P(t)}. \end{aligned} \tag{8}$$

4.3 Maximum Predictability Π^{max} under Context-Transition Entropy

The predictability Π is defined as the success rate of predicting the next location accurately given the trajectory sequence. The limits of predictability vary with the values of entropy: the Shannon entropy models the underlying probability distribution of individual locations; the real entropy models the sequential patterns of trajectories and computes the entropy of the location sequences; our proposed Context-Transition entropy considers both the context information and the location transition patterns to compute the entropy. Here we adopt Fano's inequality to connect the entropy value E of a trajectory with the upper bound of predictability Π^{max}. Specifically, the maximum predictability Π^{max} could be computed using the entropy value E of a user's trajectories and the number ($N^{(u)}$) of unique locations visited by u:

$$\begin{aligned} E = & - \Pi^{max} \log_2(\Pi^{max}) - (1 - \Pi^{max}) \log_2(1 - \Pi^{max}) \\ & + (1 - \Pi^{max}) \log_2(N^{(u)} - 1). \end{aligned} \tag{9}$$

A full proof of Equation (9) inspired by Zhao et al. [38] and Song et al. [28] is provided, which is detailed in the appendix.

If a user u has an entropy value $E = 0$, this user's next visited location will be completely regular, meaning that it is fully predictable. If, however, u has entropy $E = log_2(N^{(u)})$, the next visited location will be expected to follow a totally random pattern. In this circumstance, the next location cannot be predicted with accuracy that exceeds $1/N^{(u)}$. Actually, most users have finite entropy values between 0 and $log_2(N^{(u)})$, indicating that there is some regularity in their trajectories that can be exploited for predictive purposes.

Based on users' historical trajectories, we investigate the limits of predictability of next location prediction. To be specific, we calculate the maximum predictability Π^{max} via solving Equation (9), i.e., finding the solution of Π^{max} based on the Householder's method [30]. We have different measures for the maximum predictability: $\Pi^{Shannon}(\mathcal{E}_u^l)$, $\Pi^{Real}(\mathcal{E}_u^l)$, and $\Pi_{Transition}^{Context}(\mathcal{E}_u^l | \mathcal{E}_u^t)$, corresponding to various entropy measures: the Shannon entropy $E^{Shannon}(\mathcal{E}_u^l)$, the real entropy $E^{Real}(\mathcal{E}_u^l)$, and the Context-Transition entropy $E_{Transition}^{Context}(\mathcal{E}_u^l | \mathcal{E}_u^t)$. As the real entropy captures the sequential patterns of the location sequences compared to the Shannon Entropy, thus $E^{Shannon}(\mathcal{E}_u^l) \geq E^{Real}(\mathcal{E}_u^l)$. Since our proposed Context-Transition entropy $E_{Transition}^{Context}(\mathcal{E}_u^l | \mathcal{E}_u^t)$ is a conditional entropy on both contextual information and transition patterns, naturally, for each user, we obtain that $E^{Real}(\mathcal{E}_u^l) \geq E_{Transition}^{Context}(\mathcal{E}_u^l | \mathcal{E}_u^t)$. Thus our proposed Context-Transition entropy is the smallest among all entropy measures: $E^{Shannon}(\mathcal{E}_u^l) \geq E^{Real}(\mathcal{E}_u^l) \geq$

$E_{Transition}^{Context}(\mathcal{E}_u^l|\mathcal{E}_u^t)$. It has been proven that the Equation (9) is concave and monotonically decreases with the entropy [28]. When the entropy is approaching 0, the solution to Equation (9) approaches 1 [38]. Thus we obtain $\Pi^{Shannon}(\mathcal{E}_u^l) \leq \Pi^{Real}(\mathcal{E}_u^l) \leq \Pi_{Transition}^{Context}(\mathcal{E}_u^l|\mathcal{E}_u^t)$, making $\Pi_{Transition}^{Context}(\mathcal{E}_u^l|\mathcal{E}_u^t)$ to be the most accurate approximation of maximum predictability Π^{max}. Therefore, $\Pi_{Transition}^{Context}(\mathcal{E}_u^l|\mathcal{E}_u^t)$ is Π^{max} in our study.

Now, according to the value of $\Pi_{Transition}^{Context}(\mathcal{E}_u^l|\mathcal{E}_u^t)$, we are able to answer the Problem 1. The $\Pi_{Transition}^{Context}(\mathcal{E}_u^l|\mathcal{E}_u^t)$ is a value ranging from 0 to 1. If the value of $\Pi_{Transition}^{Context}(\mathcal{E}_u^l|\mathcal{E}_u^t)$ is larger, we could obtain a larger forecasting accuracy with the prediction algorithm. For most users, the next location they visit is governed by a certain amount of randomness (e.g. irregular events, a car accident) and some degree of regularity when considering the contextual information (e.g. weekly patterns, go home during 4 p.m. and 6 p.m. on weekdays), which can be exploited for prediction. For example, a user with Π^{max} = 0.8 indicates that for about 20% of the time their trip appears to be totally random. In other words, no matter how good the predictive algorithm is, we cannot forecast the future location of a user with $\Pi^{max} = 0.8$ with an accuracy that is better than 80%. $\Pi_{Transition}^{Context}(\mathcal{E}_u^l|\mathcal{E}_u^t)$ represents the fundamental limit of predictability for human mobility prediction. We empirically analyze the maximum predictability in detail in Section 6.

5 Next Location Predictors

We employ four commonly-used predictive algorithms for forecasting the next locations: the Bayes model (a probability-based algorithm), the matrix factorization model (a preference-based algorithm), the deep recurrent neural network (a sequential-based algorithm), and the embedding learning model (a context-based algorithm). In this section, we answer the following question:

Problem 2. *Given the Context-Transition Predictability as the limit bound, we aim to find which predictive algorithm is able to approach the newly defined limits of predictability.*

5.1 The Bayes Predictor

Assume that a user u has a trajectory $T_i = \langle (l_1, t_1), \dots, (l_i, t_i) \rangle$ and currently locates at l_i. We aim to compute the probability $P(l_{i+1}|T_i)$ for each candidate location l_{i+1} that u will visit next given T_i. Intuitively, people's next visited location is affected by multiple factors, e.g., the just visited location, the current time, and the individual preference [3], [27]. Specifically, we make three assumptions: first, the next location l_{i+1} mainly depends on the previous location l_i according to the memorylessness feature of the Markov chains; second, users usually have different next location choices at different time, e.g., a user is leaving home, and she/he will be more likely to go to work in the morning and visit a shopping mall in the evening; finally, users have their own preference on locations due to different habits and behavior patterns. Thus, we model the local sequential transition, the temporal factor, and the user preference in predicting the next location, and define the probability $P(l_{i+1}|T_i)$ as follows:

$$P(l_{i+1}|T_i) = P(l_{i+1}|l_i, h_i, u), \tag{10}$$

where h_i is the hour of day that the visit of location l_i occurs. Essentially, this approach gauges the probability that a user would visit each candidate location l_{i+1}, and opts for the one that has the highest possibility as the predicted one.

To calculate the probability $P(l_{i+1}|T_i)$, we assume that l_i, h_i, and u are conditionally independent with l_{i+1} and re-define the probability in Equation (10) as follows:

$$\begin{aligned} &P(l_{i+1}|l_i, h_i, u) = \\ &\qquad \frac{P(l_i|l_{i+1})P(h_i|l_{i+1})P(u|l_{i+1})P(l_{i+1})}{\sum_{l_{i+1}} P(l_i|l_{i+1})P(h_i|l_{i+1})P(u|l_{i+1})P(l_{i+1})}. \end{aligned} \tag{11}$$

To this end, we instead calculate the conditional probabilities $P(l_i|l_{i+1})$, $P(h_i|l_{i+1})$, and $P(u|l_{i+1})$. We adopt the commonly-used maximum likelihood principle to calculate the values of these conditional probabilities:

$$\begin{aligned} P(l_i|l_{i+1}) &= \frac{\sharp(l_i, l_{i+1})}{\sharp(l_{i+1})}, \\ P(h_i|l_{i+1}) &= \frac{\sharp(h_i, l_{i+1})}{\sharp(l_{i+1})}, \\ P(u|l_{i+1}) &= \frac{\sharp(u, l_{i+1})}{\sharp(l_{i+1})}, \end{aligned} \tag{12}$$

where $\sharp(l_i, l_{i+1})$ denotes the number of times that l_{i+1} occurs after l_i immediately in a trajectory, $\sharp(h_i, l_{i+1})$ denotes the number of times that a user visits l_{i+1} in hour h_i, $\sharp(u, l_{i+1})$ denotes the number of times that user u visits l_{i+1}, and $\sharp(l_{i+1})$ is the number of times that location l_{i+1} occurs in the trajectory set. Based on Equations (11) and (12), we compute the value of $P(l_{i+1}|T_i)$ for each candidate location l_{i+1}, and choose l_{i+1} with the largest value as the predicted next location.

5.2 The Weighted Matrix Factorization Predictor

Given users' trajectories, we first construct a user-location preference matrix $\mathbf{R} = [r_{u,l}] \in \{0,1\}^{N_u \times N_l}$, where N_u is the total number of users and N_l is the total number of locations. If a user u has visited a location l, $r_{u,l}$ will be 1. Similarly, we also construct a user-location visiting frequency matrix $\mathbf{F} \in \mathbb{R}^{N_u \times N_l}$, where $f_{u,l}$ indicates the frequency of a user u visiting a location l. Using the preference matrix $\mathbf{R}$, location prediction could be regarded as the One-Class Collaborative Filtering problem [11], [18]. Specifically, for a user, we treat her/his visited locations as the positive and all unvisited locations as the negative. As the visiting frequency of a user is a confidence of her/his positive preference, we define a weighting matrix $\mathbf{W} = [w_{u,l}]$ which assigns different weights to the positive and the negative locations based on the visiting frequency matrix $\mathbf{F}$. That is, $w_{u,l}$ is defined as

$$w_{u,l} = \begin{cases} 1 + \alpha f_{u,l} & \text{if } f_{u,l} > 0 \\ 1 & \text{otherwise} \end{cases}, \tag{13}$$

where α is a positive constant.

Using the weighting matrix, we minimize the objective of the Weighted Matrix Factorization (WMF) predictor:

$$\sum_{u,l} w_{u,l} \left(r_{u,l} - \vec{v}_u \cdot \vec{v}_l \right)^2, \tag{14}$$

where $\vec{v}_u$ and $\vec{v}_l$ are the feature vectors of a user u and a location l, respectively. Note we use the regularization term to prevent overfitting. This objective maps users and locations into a latent semantic space, where a user's preference for a location is modeled as the inner product between them.

5.3 The LSTM Predictor

As the trajectory data is a typical sequential data, it is natural to utilize the LSTM [10] to model these trajectories for location prediction. The LSTM predictor is good at capturing the long-term sequential patterns inherent in trajectories, and performs better than some other location predictors in the task of next check-in point-of-interest prediction as stated in [22]. Here we implement a simple LSTM predictor for modeling the trajectories. Given a trajectory T_i, we input the location sequence $\langle l_1, \cdots, l_k, \cdots, l_i \rangle$ into the LSTM cells which contain input, output, cell and forget gates. The specific calculation is as follows:

$$\begin{aligned} i_k &= \sigma(W_i \vec{v}_{l_k} + U_i s_{k-1}) \\ f_k &= \sigma(W_f \vec{v}_{l_k} + U_f s_{k-1}) \\ g_k &= \tanh(W_g \vec{v}_{l_k} + U_g s_{k-1}) \\ o_k &= \sigma(W_o \vec{v}_{l_k} + U_o s_{k-1}) \\ c_k &= f_k \odot c_{k-1} + i_k \odot g_k \\ s_k &= o_k \odot \tanh(c_k). \end{aligned} \tag{15}$$

i_k, f_k, g_k, o_k are the input, forget, cell, and output gates, respectively. $\vec{v}_{l_k}$ denotes the embedding vector of the location l_k, s_{k-1} is the hidden state at time t_{k-1}, W and U are the weight matrices, c_{k-1} and c_k are the cell states at time t_{k-1} and t_k. σ and $\odot$ represent the sigmoid function and the Hadamard product, respectively. All these gates consider both the current input $\vec{v}_{l_k}$ and the previous hidden state s_{k-1} by weight matrices and activation functions. To obtain the new cell state c_k, f_k is pointwise multiplied with the previous cell state c_{k-1}, and the resulting product is summed with the pointwise product of i_k and g_k. Finally, to yield the new hidden state s_k, o_k is pointwise multiplied with the output of a tanh applied on c_k. In the stage of model training, we utilize the softmax function to calculate the output of each time step, and then use the cross entropy loss function to calculate its error with the output. The final output will be $\langle l_2, l_3, \ldots, \hat{l}_{i+1} \rangle$ and $\hat{l}_{i+1}$ is the next location predicted by the LSTM model.

5.4 The Embedding Predictor

We design an embedding predictor for location prediction, which models both the sequential patterns and the contextual information of trajectories. Assume that a user u has a trajectory $T_i = \langle (l_1, t_1), \cdots, (l_i, t_i) \rangle$. For each location l_k, we aim at maximizing the probability that it occurs given the corresponding context $\vec{v}_{c_k}$:

$$\ell(T_i) = \sum_{k=1}^{i} \log P(l_k | \vec{v}_{c_k}), \tag{16}$$

where $\vec{v}_{c_k}$ is the context vector of l_k. For each target location l_k, we represent it using a D-dimensional vector $\vec{v}'_{l_k}$. To compute $P(l_k | \vec{v}_{c_k})$, we assume that l_k depends on its context, and apply a multi-classifier to generate l_k based on the context vector $\vec{v}_{c_k}$ using a softmax function:

$$P(l_k | \vec{v}_{c_k}) = \frac{\exp(\vec{v}_{c_k} \cdot \vec{v}'_{l_k})}{\sum_{l \in \mathcal{L}^t} \exp(\vec{v}_{c_k} \cdot \vec{v}'_l)}, \tag{17}$$

where $\mathcal{L}^t$ is the set of locations.

In this predictor, the context consists of the sequential context, the temporal context, and the personal context. To construct the sequential context, we assume that the location l_k is mainly associated with the previous visited M locations $(l_{i-M}, \cdots, l_{i-1})$ in the trajectory. We also utilize a D-dimensional vector $\vec{v}_l$ to represent each context location. Note that the representations of the target locations and the context locations are different. Using these vectors of context locations, the vector of the sequential context is represented as $\sum_{m=1}^{M} \vec{v}_{l_{i-m}} / M$.

Further, we represent the time slot h (e.g., an hour) with a D-dimensional vector and build the temporal context for the target location l_k. As the users' next location choices are usually influenced by the time of current visits, the temporal context of l_k is defined as the latest time $\vec{v}_{h_{k-1}}$. Finally, we integrate the three kinds of context and build $\vec{v}_{c_k}$ according to the additivity assumption [3], [43],

$$\vec{v}_{c_k} = \frac{1}{3} \left(\frac{1}{M} \sum_{m=1}^{M} \vec{v}_{l_{k-m}} + \vec{v}_{h_{k-1}} + \vec{v}_u \right), \tag{18}$$

where $\vec{v}_u$ is the D-dimensional vector of user u. In addition to the sum aggregation method, other aggregation methods (e.g., one-dimensional convolution) could also be utilized.

Given the trajectories of all the users, we maximize the overall objective function ℓ:

$$\ell = \sum_{u \in \mathcal{U}} \sum_{T \in \mathcal{T}_u} \sum_{k=1}^{N_T} \log P(l_k | \vec{v}_{c_k}), \tag{19}$$

where $\mathcal{U}$ represents the user set, $\mathcal{T}_u$ represents all the trajectories of u, and N_T is the number of locations in trajectory T. For parameter learning, we adopt the method of negative sampling [3] and learn all parameters with the Stochastic Gradient Descent method.

After obtaining the embedding vectors of locations, time slots and users, we utilize them to predict the next locations. Given a test trajectory T_i, we first construct the contextual vector $\vec{v}_c$ according to Equation (18). Then, for each candidate location $l \in \mathcal{L}^t$, we compute $P(l|\vec{v}_c)$ using the following function:

$$P(l|\vec{v}_c) \propto \left(\frac{1}{M} \sum_{m=1}^{M} \vec{v}_{l_{i-m+1}} + \vec{v}_{h_i} + \vec{v}_u \right) \cdot \vec{v}'_l. \tag{20}$$

Finally, we sort the values of $P(l|\vec{v}_c)$ and select the top location as the predicted one.

6 Increased Upper Bound with Context-Transition Predictability

We first introduce the datasets we used in the experiment, and then analyze the predictability of location prediction.

6.1 Datasets

The users' check-in data from Foursquare collected from New York City and Tokyo are public datasets [33], [34], ranging from April 2012 to September 2013. Each record contains the following four attributes: user ID, check-in location ID, location category, and check-in time. A sample

TABLE 2: Examples of the NYC dataset.

user ID	location ID	location category	check-in time
728	49f68bb6f964a5204f6c1fe3	Bar	Tue Apr 03 2012 23:34:14
728	43bba61df964a520eb2c1fe3	Food & Drink Shop	Eat Apr 07 2012 16:44:42
728	4a452221f964a520d8a71fe3	Beer Garden	Eat Apr 07 2012 23:15:37
728	4db84a97f7b15ca52ce70e7f	Office	Mon Apr 09 2012 09:13:27
728	48778b89f964a52014511fe3	Bar	Mon Apr 09 2012 21:31:19
728	4ab04eb9f964a520fe6620e3	Hotel	Tue Apr 17 2012 21:23:33

TABLE 3: Data statistics.

	NYC	TKY
♯users	951	1948
♯locations	4292	6994
avg. ♯check-ins per user	111	163

of the NYC (short for New York City) data is shown in Table 2. In the experiments, we filter out those locations and users whose number of check-ins is fewer than 20. We sort all check-ins of a user according to the time-stamp and generate a single trajectory for every user. After preprocessing, the statistics of the two datasets are reported in Table 3. ♯users is the number of unique users, ♯locations is the number of unique locations, and avg. ♯check-ins per user represents the number of check-in records of a user on average. We observe that for each user, the average number of locations that a user visits is 111 for the NYC dataset and 163 for the TKY dataset, which means that a random guess predictor will yield low prediction accuracy due to the sparsity of the dataset.

6.2 Increased Upper Bound of Predictability

We first compute the entropy (including the Shannon entropy, the real entropy, and the newly proposed Context-Transition entropy) and the corresponding predictability for every user using the check-in trajectories. Then we calculate the distributions of the three kinds of entropy and predictability over all these users. We report the distributions of the entropy and the predictability in Fig. 2. Specifically, Fig. 2 (a) and (c) shows the entropy distributions of the Context-Transition entropy $E^{Context}_{Transition}(\mathcal{E}^l_u|\mathcal{E}^t_u)$, the real entropy $E^{Real}(\mathcal{E}^l_u)$, and the Shannon entropy $E^{Shannon}(\mathcal{E}^l_u)$ on the NYC and TKY data. The distributions of the Context-Transition Predictability $\Pi^{Context}_{Transition}$, the real Predictability Π^{Real}, and the Shannon Predictability $\Pi^{Shannon}$ on the NYC and TKY data are reported in Fig. 2 (b) and (d) respectively. Several interesting trends could be observed from Fig. 2.

First, we observe that $E^{Shannon}(\mathcal{E}^l_u) \geq E^{Real}(\mathcal{E}^l_u) \geq E^{Context}_{Transition}(\mathcal{E}^l_u|\mathcal{E}^t_u)$, which is consistent with our proof in Section 4. The real entropy peaks at a smaller value ($E^{Real}(\mathcal{E}^l_u) = 1.87$) than the Shannon entropy ($E^{Shannon}(\mathcal{E}^l_u) = 3.74$), as it captures the sequential patterns of location sequences. The proposed Context-Transition entropy peaks at the smallest value, as it is a conditional entropy based on both contextual information and transition patterns. The small value of the Context-Transition entropy indicates a high regular pattern in these trajectories when considering both transition and contextual patterns. Further, we observe that both Π^{Real} and $\Pi^{Shannon}$ are smaller than $\Pi^{Context}_{Transition}$, i.e., $\Pi^{Shannon}(\mathcal{E}^l_u) \leq \Pi^{Real}(\mathcal{E}^l_u) \leq \Pi^{Context}_{Transition}(\mathcal{E}^l_u|\mathcal{E}^t_u)$, making $\Pi^{Context}_{Transition}(\mathcal{E}^l_u|\mathcal{E}^t_u)$ be the most accurate approximation of maximum predictability Π^{max}. The reason lies in that capturing the temporal and sequential patterns of trajectories in the predictive algorithm could bring a higher potential predictive accuracy. Our proposed Context-Transition Predictability extends the limit bound for human mobility data prediction.

TABLE 4: Effect of orders on the Context-Transition Predictability (in percentage).

	NYC	TKY
$\Pi^{Context}_{Transition}$	82.78	77.58
$\Pi^{Context}_{Order-2}$	81.13	76.24
$\Pi^{Context}_{Order-3}$	80.61	75.71
$\Pi^{Context}_{Order-4}$	80.47	75.55
$\Pi^{Context}_{Order-5}$	80.45	75.51

6.3 Effects of the Longer-term Dependencies on Context-Transition Predictability

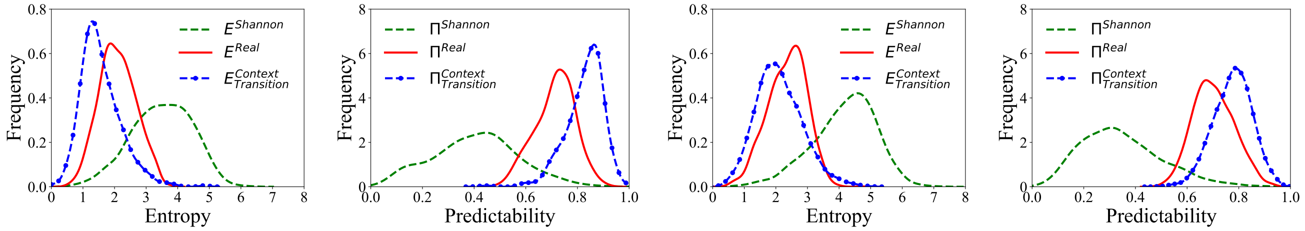
In Section 4.2, we model the location transitions to capture the sequential patterns inherent in trajectories. Here we explore whether modeling longer-term dependencies in the trajectory could improve the predictability. We consider the prefix sequences from order-2 to order-5, and report the predictability in Table 4. Evidently, with the increasing length of prefix sequences, the average values of Context-Transition Predictability are decreasing. It implies that users would like to have regular short-term transitions in their check-in trajectories. The prediction accuracy will decrease with the long-term transition sequence being predicted even considering the contextual information.

7 Evaluation: Approaching the Limits of Predictability

To validate which predictive algorithm is able to approach the newly defined limits of predictability $\Pi^{Context}_{Transition}$, we evaluate several predictors including the Random predictor, the MostPopular predictor, the UCF predictor, the WMF predictor, the Bayes predictor, the LSTM predictor, and the embedding predictor.

7.1 Experimental Setup

For each user, we take the first 80% of check-ins of her/his trajectory as the training set and the remaining 20% as the test set. We use the standard metric of $Acc@K$ where $K \in \{5, 10, 20, 30, 50\}$ to evaluate the performance of these



(a) Entropy (NYC)

(b) Predictability (NYC)

(c) Entropy (TKY)

(d) Predictability (TKY)

Fig. 2: Distributions of (a) Entropy (NYC), (b) Predictability (NYC), (c) Entropy (TKY), (d) Predictability (TKY) over all these users. The blue line, the red line, and the green line refer to the probability density function of $E^{Context}_{Transition}(\mathcal{E}^l_u|\mathcal{E}^t_u)$, $E^{Real}(\mathcal{E}^l_u)$, $E^{Shannon}(\mathcal{E}^l_u)$ and $\Pi^{Context}_{Transition}$, Π^{Real}, $\Pi^{Shannon}$, respectively.

TABLE 5: Performance in terms of $Acc@K$ on the NYC data (in percentage).

Method	NYC						
	Acc@5	Acc@10	Acc@20	Acc@30	Acc@50	Average	Relative Improvement
Random	0.07	0.12	0.29	0.53	0.98	0.40	-
MostPopular	3.68	5.31	7.00	8.04	10.25	6.86	1615.00
UCF	6.79	9.21	12.82	15.72	19.85	12.88	3120.00
WMF	14.45	22.39	29.69	33.42	37.53	27.49	6772.50
LSTM	14.12	17.94	21.64	23.76	26.99	20.89	5122.50
Bayes 24h	15.53	18.73	20.50	20.96	21.29	19.40	4750.00
Bayes 48h	15.26	18.53	20.40	20.90	21.26	19.27	4717.50
Embedding 24h	**16.38**	**25.90**	**36.00**	**40.05**	**42.12**	**32.09**	**7922.50**
Embedding 48h	15.72	25.37	35.40	39.92	42.12	31.71	7827.50

TABLE 6: Performance in terms of $Acc@K$ on the TKY data (in percentage).

Method	TKY						
	Acc@5	Acc@10	Acc@20	Acc@30	Acc@50	Average	Relative Improvement
Random	0.04	0.07	0.17	0.38	0.57	0.25	-
MostPopular	6.78	9.81	12.44	15.24	19.04	12.66	4964.00
UCF	9.91	14.00	18.65	21.89	26.58	18.21	7184.00
WMF	10.95	17.63	25.46	30.48	36.87	24.28	9612.00
LSTM	12.31	17.90	24.36	28.69	34.35	23.52	9308.00
Bayes 24h	**17.23**	**22.95**	27.79	30.28	33.03	26.26	10404.00
Bayes 48h	16.87	22.57	27.60	30.05	32.75	25.97	10288.00
Embedding 24h	12.85	20.57	**30.63**	**36.61**	**42.32**	**28.60**	**11340.00**
Embedding 48h	12.87	20.43	30.59	36.62	42.26	28.55	11320.00

predictors. Given a test sample, for $Acc@K$, if the ground truth location is within the top K of the prediction set, then a score of 1 is awarded, and else 0.

For the Bayes predictor, we consider two variants: one (named Bayes 48h) distinguishes time bins in weekdays from weekends and obtains 48 one-hour time bins (24 for weekdays and 24 for weekends) in total, and the other (named Bayes 24h) obtains 24 one-hour bins without differentiating weekdays from weekends. Similarly, we also design two variants for the embedding predictor, named Embedding 48h and Embedding 24h, respectively. In addition to the Bayes, WMF, LSTM, and embedding predictors, we also use the following three methods, Random, MostPopular, and UCF [35] as baselines for comparison. The Random predictor means that we randomly select a location from the training set as the next location. The MostPopular predictor ranks these locations using frequencies across the training set and selects the top ones as the prediction. The UCF utilizes the preference of similar users for location prediction, where the similarity between users is related to the number of their common visited locations.

7.2 Performance of Different Predictors

We show the accuracy of these predictors on both datasets in Tables 5 and 6. The best results are reported in bold. We observe that the Random predictor performs the worst (average 0.4%) as expected. This is mainly because that both datasets have a large number of locations (4,292 locations for NYC and 6,994 locations for TKY). There is a very high chance of not predicting the next locations accurately if we randomly select one of them as the next location.

The MostPopular predictor improves the prediction accuracy significantly. The popularity of locations of people's visits is not uniformly distributed and people tend to visit a few locations that they frequently visit [39]. For example, for the NYC dataset, the top 50 most visited locations out of total 4,292 locations occupy 10% of total visits. Thus the prediction accuracy has been significantly improved while using most visited places as predicted locations. The UCF predictor improves the prediction accuracy compared to the MostPopular predictor, as it utilizes more useful user-location interaction information to capture the preference of similar users and is able to address the problem of sparsity.

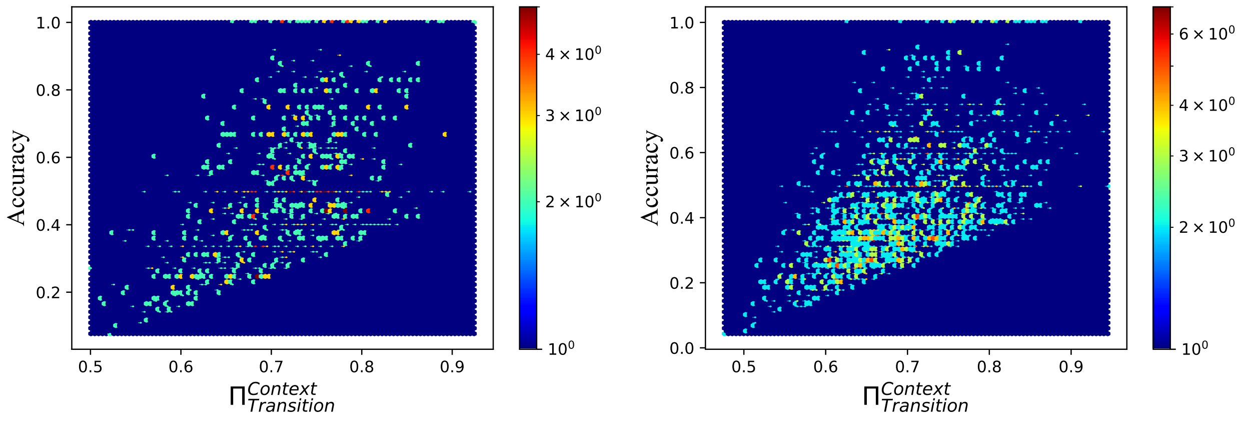


Fig. 3: The correlation heatmap between Context-Transition Predictability and prediction accuracy of Embedding 24h on the two datasets (NYC left, Tokyo right).

The WMF predictor works better for collaborative filtering from implicit users' feedback than the simple UCF.

The LSTM predictor is able to extract the long-term temporal patterns of users' historical visited locations, and it gets decent performance considerably. When the location sequence contains rich temporal correlations, LSTM can predict the next location well. However, since the average number of locations in each trajectory is high (see Table 3), the prediction accuracy reduces significantly with the long sequence of visited locations [31]. The Bayes predictor is an easy and efficient method. When predicting the next location, it considers the previous location given the conditional user and time information. Human movements have strong temporal and user-specific patterns [39]. People tend to visit a few locations given time. Thus we observe that the Bayes predictors have decent performance, for some cases (e.g., $Acc@5$ and $Acc@10$ for the TKY data) even the best predictor among all these predictors.

In the end, we observe that the embedding predictor (i.e., Embedding 24h) performs the best in most cases. The embedding predictor considers the context patterns including both user and temporal information in the training. Thus it provides the best prediction performance, as what we have expected given our Context-Transition Predictability has shown for the dataset.

The evaluation of predictors above reveals that, for this users' check-in dataset, the maximum predictability $\Pi^{Context}_{Transition}$ can be achieved with an embedding predictor. To further analyze the correlation between the embedding predictor and our proposed Context-Transition Predictability, for each user in the two datasets, we obtain their prediction accuracy under the embedding predictor and their corresponding Context-Transition Predictability, and examine the correlation between them. Here we choose the Embedding 24h predictor, as it captures both the sequential patterns and the contextual information. As we can see from Fig. 3, the prediction accuracy of the embedding predictor is highly correlated with each user's Context-Transition Predictability $\Pi^{Context}_{Transition}$; the larger the Context-Transition Predictability is, the higher the prediction accuracy is. The correlation coefficient between the prediction accuracy of the embedding predictor and the Context-Transition Predictability is as high as 0.539 for the NYC dataset, with $p < 0.01$, and 0.541 for the TKY dataset, with $p < 0.01$.

The high correlation between the Context-Transition Predictability and the prediction accuracy of the embedding predictor reveals that, $\Pi^{Context}_{Transition}$ which considers both sequential regularity and context information, is able to provide an upper bound of predictive power of human mobility data. Further, we could select human mobility predictive algorithms according to the Context-Transition Predictability. The prediction of human mobility is heavily dependent on the context information such as the check-in time and the categories of locations; the embedding predictor that models the context information under the guidance of the Context-Transition Predictability could approach the limits of predictability.

8 Discussion

8.1 Impact of Data Property

We further examine the predictability with changing of unique number of locations in the training set. We rank the locations in both datasets according to the number of visits, and pick up the top 200-3,200 locations with most visits. As shown in Table 7, we observe that our Context-Transition predictability $\Pi^{Context}_{Transition}(\mathcal{E}^l_u|\mathcal{E}^t_u)$ is decreasing with the increasing of the number of unique locations. It is mainly because the more locations added into the dataset, the harder it is to predict the short-term transitions with more possible next locations, thus our Context-Transition Predictability $\Pi^{Context}_{Transition}(\mathcal{E}^l_u|\mathcal{E}^t_u)$ is decreasing with added locations. However, we observe that the real predictability $\Pi^{Real}(\mathcal{E}^l_u)$ stays quite stable, as it captures the longest subsequence, which does not change significantly over more locations added.

8.2 Impact of Data Size

To test the generality of our findings, we show the predictability and the performance of different predictors on two large-scale check-in datasets[2]. The NYC$_{large}$ data contains 876,101 check-in records of 11,098 users at 17,494

[2] https://sites.google.com/site/yangdingqi/home/foursquare-dataset

TABLE 7: The average of the three predictability on the two datasets picking top locations for different numbers (in percentage).

Top	NYC			TKY		
	$\Pi^{Shannon}$	Π^{Real}	$\Pi^{Context}_{Transition}$	$\Pi^{Shannon}$	Π^{Real}	$\Pi^{Context}_{Transition}$
200	55.62	68.73	92.70	43.00	71.57	85.95
400	53.96	71.15	91.71	40.82	71.47	84.39
800	52.63	73.43	89.71	38.92	71.13	82.81
1600	48.88	73.47	86.95	38.59	71.42	81.18
3200	42.90	72.10	83.73	37.37	71.20	79.36

locations; The TKY$_{large}$ data contains 1,331,329 check-in records of 9,552 users at 13,928 locations. We compute the entropy and the corresponding predictability for every user using the two large datasets, and report the distributions of the entropy and the predictability in Supplementary Figure S1. We also report the accuracy of these predictors on both datasets in Supplementary Tables SI and SII. We have similar findings as the one we show in the evaluation section: the proposed Context-Transition entropy peaks at the smallest value, and $\Pi^{Context}_{Transition}$ is the most accurate approximation of maximum predictability Π^{max}; the maximum predictability $\Pi^{Context}_{Transition}$ can be achieved with the embedding predictor.

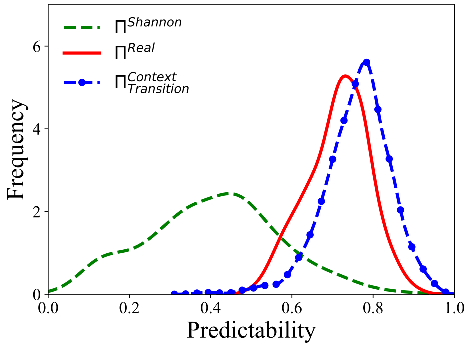


Fig. 4: Distribution of predictability given category over all these users on the NYC data.

8.3 Other Contextual Information

We also examine the proposed Context-Transition Predictability $\Pi^{Context}_{Transition}(\mathcal{E}^l_u|\mathcal{E}^t_u)$ on other contexts, namely the categorical information. There are different categories for each location, e.g., restaurant or bar information. If we know the context information (categorical information in our case) of previous visited locations, we can also have a higher chance of predicting the next locations accurately. For example, considering locations that customers usually visit after the restaurant, you might observe that the next frequently visited location will be a bar which might be far away, instead of the residential place nearby, as people usually visit a bar for a drink after dinner. Even if they are far away, they are still strongly connected with each other. As shown in Fig. 4, we find that given the categorical context information, our Context-Transition Predictability $\Pi^{Context}_{Transition}(\mathcal{E}^l_u|\mathcal{E}^t_u)$ still outperforms other predictability.

It must be noted that there is a limit to the context information of our Context-Transition Predictability. Only the contextual information of the location, such as the temporal information, the category of the location, the weather, etc can be utilized in our definition. The context information of the user such as the social relationship cannot be used to define the Context-Transition Predictability as it is fixed for each user and not changing with locations.

9 Conclusion

In this paper, we have proposed a method to compute the limits of predictability in human mobility prediction, which allows for the use of contextual information such as the time or the type of places that a person just visits inherent. Specifically, we propose an entropy estimator named Context-Transition entropy that can capture both the sequential orders of human mobility and these contextual information, to derive the limits of predictability. We show that when the context information is encoded, we get larger values for the proposed Context-Transition Predictability than the existing predictability. Further, we have reevaluated the popular location prediction methods, and demonstrated that the Context-Transition Predictability is useful for the selection of predictors for next check-in location prediction. As a measurement for the sequential regularity and the context information, our proposed Context-Transition Predictability can capture the theoretical limits for human mobility prediction and guide the selection of predictive algorithms.

10 Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61906107, the Natural Science Foundation of Shandong Province of China under Grant No. ZR2019BF010, and the Young Scholars Program of Shandong University.

References

[1] B. Chang, Y. Park, D. Park, S. Kim, and J. Kang. Content-aware hierarchical point-of-interest embedding model for successive poi recommendation. In *IJCAI*, pages 3301–3307, 2018.

[2] M. Chen, X. Yu, and Y. Liu. Mining moving patterns for predicting next location. *IS*, 54:156–168, 2015.

[3] M. Chen, X. Yu, and Y. Liu. Mpe: a mobility pattern embedding model for predicting next locations. *WWWJ*, pages 1–20, 2018.

[4] M. Chen, Y. Zhao, Y. Liu, X. Yu, and K. Zheng. Modeling spatial trajectories with attribute representation learning. *TKDE*, 2020.

[5] A. Cuttone, S. Lehmann, and M. C. González. Understanding predictability and exploration in human mobility. *EPJ DATA SCI*, 7:1–17, 2018.

[6] A. De Brébisson, É. Simon, A. Auvolat, P. Vincent, and Y. Bengio. Artificial neural networks applied to taxi destination prediction. *arXiv preprint arXiv:1508.00021*, 2015.

[7] M. De Domenico, A. Lima, and M. Musolesi. Interdependence and predictability of human mobility and social interactions. *PMC*, 9(6):798–807, 2013.

[8] S. Feng, X. Li, Y. Zeng, G. Cong, and Y. M. Chee. Personalized ranking metric embedding for next new poi recommendation. In *IJCAI*, pages 2069–2075. ACM, 2015.

[9] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, June 2008.

[10] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput*, 9(8):1735–1780, 1997.

[11] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *ICDM*, pages 263–272. Ieee, 2008.

[12] S. M. Iacus, C. Santamaria, F. Sermi, S. Spyratos, D. Tarchi, and M. Vespe. Human mobility and covid-19 initial dynamics. *Nonlinear Dyn.*, 101(3):1901–1919, 2020.

[13] D. Kong and F. Wu. Hst-lstm: A hierarchical spatial-temporal long-short term memory network for location prediction. In *IJCAI*, pages 2341–2347, 2018.

[14] I. Kontoyiannis, P. H. Algoet, Y. M. Suhov, and A. J. Wyner. Nonparametric entropy estimation for stationary processes and random fields, with applications to English text. *TIT*, 44(3):1319–1327, 1998.

[15] J. R. Koo, A. R. Cook, M. Park, Y. Sun, H. Sun, J. T. Lim, C. Tam, and B. L. Dickens. Interventions to mitigate early spread of sars-cov-2 in singapore: a modelling study. *Lancet Infect Dis*, 20(6):678–688, 2020.

[16] M. U. Kraemer, C.-H. Yang, B. Gutierrez, C.-H. Wu, B. Klein, D. M. Pigott, L. Du Plessis, N. R. Faria, R. Li, W. P. Hanage, et al. The effect of human mobility and control measures on the covid-19 epidemic in china. *Science*, 368(6490):493–497, 2020.

[17] B. Li, D. Zhang, L. Sun, C. Chen, S. Li, G. Qi, and Q. Yang. Hunting or waiting? discovering passenger-finding strategies from a large-scale real-world taxi dataset. In *PERCOM Workshops*, pages 63–68, 2011.

[18] D. Lian, K. Zheng, Y. Ge, L. Cao, E. Chen, and X. Xie. Geomf++: scalable location recommendation via joint geographical modeling and matrix factorization. *TOIS*, 36(3):1–29, 2018.

[19] D. Lian, V. W. Zheng, and X. Xie. Collaborative filtering meets next check-in location prediction. In *WWW*, pages 231–232, 2013.

[20] D. Lian, Y. Zhu, X. Xie, and E. Chen. Analyzing location predictability on location-based social networks. In *PAKDD*, pages 102–113. Springer, 2014.

[21] Q. Liu, S. Wu, D. Wang, Z. Li, and L. Wang. Context-aware sequential recommendation. In *ICDM*, pages 1053–1058. IEEE, 2016.

[22] Q. Liu, S. Wu, L. Wang, and T. Tan. Predicting the next location: A recurrent model with spatial and temporal contexts. In *AAAI*, pages 194–200, 2016.

[23] X. Lu, E. Wetter, N. Bharti, A. J. Tatem, and L. Bengtsson. Approaching the limit of predictability in human mobility. *Sci. Rep.*, 3(1):1–9, 2013.

[24] M. Luca, G. Barlacchi, B. Lepri, and L. Pappalardo. A survey on deep learning for human mobility. *CSUR*, 55(1):1–44, 2021.

[25] J. Lv, Q. Li, Q. Sun, and X. Wang. T-conv: A convolutional neural network for multi-scale taxi trajectory prediction. In *BigComp*, pages 82–89. IEEE, 2018.

[26] S. Qiao, N. Han, J. Wang, R.-H. Li, L. A. Gutierrez, and X. Wu. Predicting long-term trajectories of connected vehicles via the prefix-projection technique. *TITS*, 19(7):2305–2315, 2017.

[27] Y. Qiao, Z. Si, Y. Zhang, F. B. Abdesslem, X. Zhang, and J. Yang. A hybrid markov-based model for human mobility prediction. *Neurocomputing*, 278:99–109, 2018.

[28] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, 2010.

[29] D. d. C. Teixeira, A. C. Viana, M. S. Alvim, and J. M. Almeida. Deciphering predictability limits in human mobility. In *SIGSPATIAL*, pages 52–61, 2019.

[30] J. H. Wilkinson and J. H. Wilkinson. *The algebraic eigenvalue problem*, volume 87. Clarendon Press Oxford, 1965.

[31] Y. Xiao and Q. Nian. Vehicle location prediction based on spatiotemporal feature transformation and hybrid lstm neural network. *Information*, 11(2):84, 2020.

[32] C. Xiong, S. Hu, M. Yang, W. Luo, and L. Zhang. Mobile device data reveal the dynamics in a positive relationship between human mobility and covid-19 infections. *PNAS*, 117(44):27087–27089, 2020.

[33] D. Yang, D. Zhang, L. Chen, and B. Qu. Nationtelescope: Monitoring and visualizing large-scale collective behavior in lbsns. *JNCA*, 55:170–180, 2015.

[34] D. Yang, D. Zhang, and B. Qu. Participatory cultural mapping based on collective behavior data in location-based social networks. *TIST*, 7(3):30, 2016.

[35] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*, pages 325–334, 2011.

[36] H. Ying, J. Wu, G. Xu, Y. Liu, T. Liang, X. Zhang, and H. Xiong. Time-aware metric embedding with asymmetric projection for successive poi recommendation. *WWWJ*, 22(5):2209–2224, 2019.

[37] R. Zhang and M. Pavone. Control of robotic mobility-on-demand systems: A queueing-theoretical perspective. *IJRR*, 35(1-3):186–203, 2016.

[38] K. Zhao, D. Khryashchev, and H. Vo. Predicting taxi and uber demand in cities: Approaching the limit of predictability. *TKDE*, 33(6):2723–2736, 2021.

[39] K. Zhao, S. Tarkoma, S. Liu, and H. Vo. Urban human mobility data mining: An overview. In *Big Data*, pages 1911–1920. IEEE, 2016.

[40] K. Zhao, Y. Zhang, H. Yin, J. Wang, K. Zheng, X. Zhou, and C. Xing. Discovering subsequence patterns for next poi recommendation. In *IJCAI*, pages 3216–3222, 2020.

[41] P. Zhao, A. Luo, Y. Liu, F. Zhuang, J. Xu, Z. Li, V. S. Sheng, and X. Zhou. Where to go next: A spatio-temporal gated network for next poi recommendation. *TKDE*, 2020.

[42] S. Zhao, T. Zhao, H. Yang, M. R. Lyu, and I. King. Stellar: Spatial-temporal latent ranking for successive point-of-interest recommendation. In *AAAI*, 2016.

[43] N. Zhou, W. X. Zhao, X. Zhang, J. Wen, and S. Wang. A general multi-context embedding model for mining human trajectory data. *TKDE*, 28(8):1945–1958, 2016.

Chao Zhang is a Bachelor degree student in the School of Software, Shandong University, China. He has been working in Dr. Chen's research group as a research assistant for over two years. His research interest is data mining.

Kai Zhao is an assistant professor at J. Mack Robinson College of Business, Georgia State University. His research interests are data mining, mobile computing and AI for Business. Before coming to GSU, Dr. Kai Zhao worked as a post-doc associate at the New York University 2016-2017. He received his PhD in Computer Science from University of Helsinki, Finland in 2015. Dr. Kai Zhao has authored or co-authored over 30 publications in refereed journals and prestigious conference presentations.

Meng Chen received his Ph.D. degree in computer science and technology in 2016 from Shandong University, China. He worked as a Postdoctoral fellow from 2016 to 2018 in the School of Information Technology, York University, Canada. He is currently an associate professor in the School of Software, Shandong University, China. His research interest is in the area of trajectory data mining and traffic management. He has published over 20 papers in prestigious journals and conferences in data mining field such as TKDE, TOIS, TITS and CIKM.

Appendix A Proof

Proof. Given any predictive algorithm $\mathcal{A}$, let $P_{\mathcal{A}}(l_{i+1}|T_i)$ be the distribution generated over the next possible location given the trajectory T_i. $P(l_{i+1}=l|T_i)$ is the probability that the next visited location l_{i+1} given T_i is l; $P(l_{i+1}|T_i)$ is the true distribution over the next visited location given T_i.

Let $\pi(T_i)$ be the probability that there is a most likely location given T_i. We have

$$\pi(T_i) = sup_l\{P(l_{i+1}=l|T_i)\},$$

where sup is the supremum function, the least upper bound of $P(l_{i+1}=l|T_i)$. The probability of successfully predicting the next location is $P_{\mathcal{A}}(l_{i+1}|T_i)=\sum_l P(l|T_i)P_{\mathcal{A}}(l|T_i)$. Since $\pi(T_i)\geq P(l|T_i)$ for any l, we have

$$\begin{aligned} P_{\mathcal{A}}(l_{i+1}|T_i) &= \sum_l P(l|T_i)P_{\mathcal{A}}(l|T_i) \\ &\leq \sum_l \pi(T_i)P_{\mathcal{A}}(l|T_i) \\ &= \pi(T_i)\sum_l P_{\mathcal{A}}(l|T_i) \\ &= \pi(T_i). \end{aligned} \tag{21}$$

We define the predictability of a trajectory T_i of length i as $\Pi(i)$. $P(T_i)$ represents the probability of a particular trajectory T_i. If we sum over all the possible trajectories of length i, we have the predictability as $\Pi(i) = \sum\limits_{T_i} P(T_i)\pi(T_i)$. Thus the predictability Π is defined as

$$\Pi = \lim_{i\to\infty}\frac{1}{i}\sum_{k=1}^{i}\Pi(k). \tag{22}$$

Given a trajectory T_i, we assume that a user u arrives at the next location with a probability p and the rest $(N^{(u)}-1)$ possible locations with a uniform distribution, where $N^{(u)}$ denotes the number of unique locations visited by u. Then we could get a distribution $P(l'|T_i) = (p, \frac{1-p}{N^{(u)}-1}, \frac{1-p}{N^{(u)}-1} \ldots, \frac{1-p}{N^{(u)}-1})$ for any l'. As this distribution is at least as random as the original, we know $E(l_{i+1}|T_i) \leq E(l'|T_i)$. Then we have

$$\begin{aligned} E(l'|T_i) &= -plog_2(p) - \sum\frac{1-p}{N^{(u)}-1}log_2(\frac{1-p}{N^{(u)}-1}) \\ &= -plog_2(p) - (1-p)log_2(\frac{1-p}{N^{(u)}-1}) \\ &= -[plog_2(p) + (1-p)log_2(1-p)] \\ &\quad\quad + (1-p)log_2(N^{(u)}-1) \\ &= E_F(p) \\ &= E_F(\pi(T_i)). \end{aligned} \tag{23}$$

Note here the Fano's function $E_F(p)$ is concave and monotonically decreases with p. Based on Fano's inequality, $E(l_{i+1}|T_i) \leq E_F(\pi(T_i))$. Following Jensen's inequality, we have

$$\begin{aligned} E(l_{i+1}) &= \sum_{T_i} P(T_i)E(l_{i+1}|T_i) \\ &\leq \sum_{T_i} P(T_i)E_F(\pi(T_i)) \\ &\leq E_F\left(\sum_{T_i} P(T_i)\pi(T_i)\right) \\ &= E_F(\Pi(i)). \end{aligned} \tag{24}$$

For a stationary stochastic process $\langle l_1, l_2, \ldots, l_i\rangle$, based on $E(l_{i+1}) \leq E_F(\Pi(i))$, Jensen's inequality, and Equation (22), we have

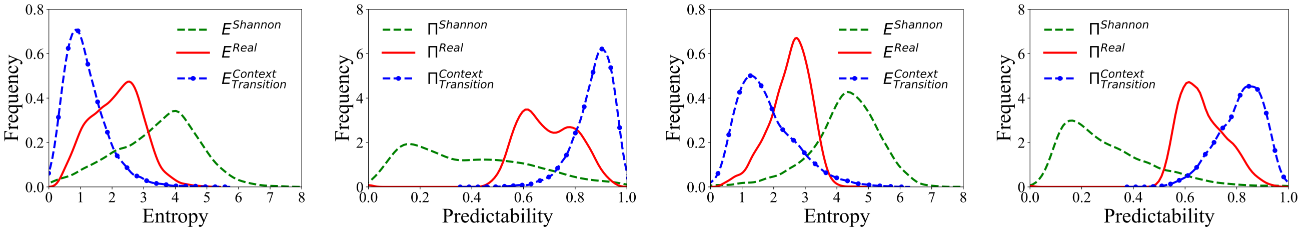
$$\begin{aligned} E &= \lim_{i\to\infty}\frac{1}{i}E(l_1, l_2, \ldots, l_i) \\ &= \lim_{i\to\infty}\frac{1}{i}\sum_{k=1}^{i}E(l_{k+1}|T_k) \\ &= \lim_{i\to\infty}\frac{1}{i}\sum_{k=1}^{i}E(l_{k+1}) \\ &\leq \lim_{i\to\infty}\frac{1}{i}\sum_{k=1}^{i}E_F(\Pi(k)) \\ &\leq E_F\left(\lim_{i\to\infty}\frac{1}{i}\sum_{k=1}^{i}(\Pi(k))\right) \\ &= E_F(\Pi). \end{aligned} \tag{25}$$

Here $E = \lim_{i\to\infty}\frac{1}{i}E(l_1, l_2, \ldots, l_i)$ is the definition of entropy. Let Π^{max} be the solution to the equation $E = E_F(\Pi^{max}) \leq E_F(\Pi)$. Since $E_F(p)$ is concave and monotonically decreased, we have $\Pi \leq \Pi^{max}$, which means that Π^{max} is the upper bound of the predictability Π. The predictability upper bound Π^{max} can be calculated by solving the following equation:

$$\begin{aligned} E &= E_F(\Pi^{max}) \\ &= -\Pi^{max}log_2(\Pi^{max}) - (1-\Pi^{max})log_2(1-\Pi^{max}) \\ &\quad\quad + (1-\Pi^{max})log_2(N^{(u)}-1). \end{aligned}$$

□

Appendix B Figures and Tables



(a) Entropy (NYC$_{large}$) (b) Predictability (NYC$_{large}$) (c) Entropy (TKY$_{large}$) (d) Predictability (TKY$_{large}$)

Supplementary Figure S1: Distributions of (a) Entropy (NYC$_{large}$), (b) Predictability (NYC$_{large}$), (c) Entropy (TKY$_{large}$), (d) Predictability (TKY$_{large}$) over all these users. The blue line, the red line, and the green line refer to the probability density function of $E^{Context}_{Transition}(\mathcal{E}^l_u|\mathcal{E}^t_u)$, $E^{Real}(\mathcal{E}^l_u)$, $E^{Shannon}(\mathcal{E}^l_u)$ and $\Pi^{Context}_{Transition}$, Π^{Real}, $\Pi^{Shannon}$, respectively.

Supplementary Table SI: Performance in terms of $Acc@K$ on the NYC$_{large}$ data (in percentage).

Method	NYC$_{large}$						
	Acc@5	Acc@10	Acc@20	Acc@30	Acc@50	Average	Relative Improvement
Random	0.02	0.03	0.06	0.14	0.26	0.10	-
MostPopular	3.74	5.90	8.45	9.82	11.96	7.97	7870.00
UCF	11.77	15.96	21.21	24.76	29.49	20.64	20540.00
WMF	9.94	14.35	19.36	22.53	26.50	18.54	18540.00
LSTM	14.69	19.73	25.20	28.82	33.25	24.34	24240.00
Bayes 24h	**19.92**	24.35	27.69	29.19	30.66	26.36	26260.00
Bayes 48h	19.44	23.89	27.39	29.03	30.52	26.05	25950.00
Embedding 24h	19.29	26.47	32.70	35.19	36.81	30.09	29990.00
Embedding 48h	19.39	**26.57**	**32.85**	**35.29**	**36.81**	**30.18**	**30080.00**

Supplementary Table SII: Performance in terms of $Acc@K$ on the TKY$_{large}$ data (in percentage).

Method	TKY$_{large}$						
	Acc@5	Acc@10	Acc@20	Acc@30	Acc@50	Average	Relative Improvement
Random	0.03	0.06	0.09	0.14	0.22	0.11	-
MostPopular	4.97	7.36	10.07	11.99	14.42	9.76	8772.73
UCF	10.36	14.31	19.09	22.32	27.05	18.63	16836.36
WMF	9.35	14.53	20.91	25.23	31.16	20.24	18300.00
LSTM	12.01	17.25	23.85	28.22	34.10	23.09	20890.91
Bayes 24h	**17.68**	**23.96**	29.52	32.34	35.54	27.81	25181.82
Bayes 48h	17.47	23.71	29.27	32.16	35.29	27.58	24972.73
Embedding 24h	13.54	21.18	**30.39**	35.41	40.37	**28.18**	**25518.18**
Embedding 48h	13.53	21.14	30.34	**35.42**	**40.41**	28.17	25509.09