

# Predicting Taxi Demand at High Spatial Resolution: Approaching the Limit of Predictability

Kai Zhao<sup>1</sup>, Denis Khryashchev<sup>3</sup>, Juliana Freire<sup>1,2</sup>, Cláudio Silva<sup>1,2</sup>, and Huy Vo<sup>1,3</sup>

<sup>1</sup>Center for Urban Science and Progress, New York University

<sup>2</sup>Department of Computer Science and Engineering, New York University

<sup>3</sup>Department of Computer Science, the City College of New York

{kai.zhao, dk2926, juliana.freire, csilva, huy.vo}@nyu.edu

**Abstract**—In big cities, taxi service is imbalanced. In some areas, passengers wait too long for a taxi, while in others, many taxis roam without passengers. Knowledge of where a taxi will become available can help us solve the taxi demand imbalance problem. In this paper, we employ a holistic approach to predict taxi demand at high spatial resolution. We showcase our techniques using two real-world data sets, yellow cabs and Uber trips in New York City, and perform an evaluation over 9,940 building blocks in Manhattan. Our approach consists of two key steps. First, we use entropy and the temporal correlation of human mobility to measure the demand uncertainty at the building block level. Second, to identify which predictive algorithm can approach the theoretical maximum predictability, we implement and compare three predictors: the Markov predictor (a probability-based predictive algorithm), the Lempel-Ziv-Welch predictor (a sequence-based predictive algorithm), and the Neural Network predictor (a predictive algorithm that uses machine learning). The results show that predictability varies by building block and, on average, the theoretical maximum predictability can be as high as 83%. The performance of the predictors also vary: the Neural Network predictor provides better accuracy for blocks with low predictability, and the Markov predictor provides better accuracy for blocks with high predictability. In blocks with high maximum predictability, the Markov predictor is able to predict the taxi demand with an 89% accuracy, 11% better than the Neural Network predictor, while requiring only 0.03% computation time. These findings indicate that the maximum predictability can be a good metric for selecting prediction algorithms.

**Index Terms**—human mobility; taxi demand prediction; spatio-temporal data; limit of predictability; predictive algorithm

## I. INTRODUCTION

In big cities, taxi service is imbalanced [1]. While in some areas passengers experience long waits for a taxi, in others, many taxis roam without passengers. This imbalance leads to profit loss for taxi companies, since vehicles are vacant even when there is demand. Besides, it reduces the level of the passenger satisfaction due to long wait times. The ability to predict taxi demand can help address the taxi-service imbalance problem. Knowledge of where a taxi should be traveling can bring benefits to both taxi drivers and companies: taxi drivers can drive to high taxi demand areas, and taxi companies (e.g. Uber) may re-allocate their vehicles in advance to meet the passenger demand.

We define the taxi-demand prediction problem as follows: given historical taxi demand data in a region  $i$ , we want to predict the number of taxis that will emerge within  $i$  during

the next time interval. Inspired by previous works [2], [3], [4], [5], [6], [7], we aim to predict the *met taxi demand*. We use the number of pick-ups as a representation of the taxi demand in a region, and treat them as sequence data (see Fig. 1). Note that although we focus on met taxi demand, our method is general and can be applied for predicting the unmet taxi demand. As we discuss in Section VII-B, unmet demand can be inferred from the met taxi demand [8], [9].

Many methods have been proposed to predict taxi demand, including uncertainty analysis [2], probabilistic models [6], time series [4], [5], SVM [3] and neural networks [7]. However, to apply these methods, we must answer two key questions: 1. Given a predictive algorithm  $\alpha$ , considering both the randomness and temporal correlation of the taxi demand sequence, what is the upper bound of the potential accuracy that a predictive algorithm  $\alpha$  can reach? 2. Given an upper bound of potential accuracy, which predictor has a better performance given the trade-off between the computation time and the accuracy?

In this paper, we answer these questions by analyzing the *maximum predictability* ( $\Pi^{max}$ ) of taxi demand in a region to select the best predictor. The maximum predictability is defined by the *entropy* of the taxi demand sequence, considering both the *randomness* and the *temporal correlation*. Maximum predictability  $\Pi^{max}$  was first introduced by Song et. al for analyzing user mobility [10]. They used a data set capturing the mobility patterns of 50,000 individuals over three months and showed that there is a potential 93% predictability in user mobility. Here, we define the maximum predictability  $\Pi^{max}$  as the highest potential accuracy that a predictive algorithm can reach for predicting taxi demand.

The maximum predictability captures the degree of temporal correlation of the taxi demand sequence by measuring the regularity of human mobility. For most regions, the taxi demand is governed by a certain amount of randomness (e.g., unexpected events) and some degree of regularity (e.g., weekly patterns), which can be exploited for prediction. For example, a building block with  $\Pi^{max} = 0.3$  indicates that for at least 70% of the time, the taxi demand of this block appears to be random, and only for 30% of the time can we hope to predict the taxi demand. In other words, no matter how good the predictive algorithm is, we cannot predict with better than 30% accuracy the future taxi demand of a building block

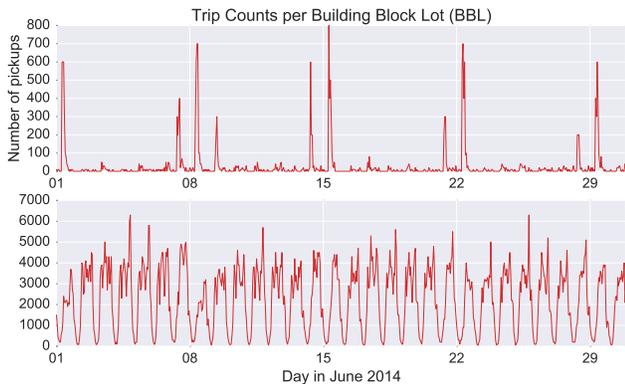


Figure 1. Heterogeneous taxi demand sequences in two building blocks: (top) a dense traffic area around the Metropolitan Museum of Art; and (bottom) a more open traffic area near the west port in NYC.

with  $\Pi^{max} = 0.3$ .  $\Pi^{max}$  represents the fundamental limit for predictability of the taxi demand in a building block.

a) *Predictability Variation in Different Regions:* Previous work assumed that the maximum predictability (the degree of the temporal correlation) in different regions is the same, and proposed the use of a single predictive algorithm for all regions [4]. However, the strong temporal correlation of taxi demand does not always hold. Different regions have different functions and thus different predictability (see Section V-B). Fig 1 shows the hourly taxi demand from two building blocks in NYC. The taxi demand near the Metropolitan Museum of Art (MoMA) (Fig.1 top) exhibits a strong temporal pattern. The regular peaks in MoMA happen during weekends and after the close time every weekend – people usually visit the museums during weekends and leave after the close time. In contrast, the taxi demand near the west port (Fig. 1 below) seems to be more random. There is no clear temporal pattern near the west port. This is because the taxi demand in a transportation hub such as the west port is heavily dependent on the arrival of ships and there is a high variability of arrival time [11]. In fact, the taxi demand near MoMA has one of the highest predictability among all the building blocks in NYC, and the taxi demand near the west port has one of the lowest predictability. Intuitively we should use different predictors for predicting the taxi demand in these two building blocks. For MoMA it is better to use a predictor that is able to capture the temporal correlation, for example, a Markov predictor. For the west port, a predictor that uses machine learning and capture additional features such as the ship schedule may be more effective. We posit that to select best predictor, we must analyze the *maximum predictability* ( $\Pi^{max}$ ) of the taxi demand in each region.

**Contributions.** In this paper, we make three key contributions:

- We measure the theoretical maximum predictability of the taxi demand for each building block in NYC. This represents the upper bound of the potential accuracy that a predictive algorithm  $\alpha$  can reach. We show that the

maximum predictability of the taxi demand can reach up to 83% on average. The maximum predictability captures the degree of the temporal correlation of the taxi demand sequence. Our findings indicate that taxi demand in NYC has a strong temporal patterns.

- We implement and compare the prediction accuracy of three predictors: the Markov predictor (a probability-based method) [12], the Lempel-Ziv-Welch (LZW) predictor (a sequence-based method) [13], and the Neural Network (NN) predictor (a machine learning-based method) [7]. We observe that the NN predictor provides better accuracy for building blocks with low predictability, and the Markov predictor provides better accuracy for building blocks with high predictability. Our findings indicate that the maximum predictability is an approachable target for actual prediction accuracy and a compute-intensive NN predictor with multiple features does not always outperform a simpler Markov predictor.
- We show that knowledge of the predictability at each building block can help determine which predictor to use, while taking accuracy and computational cost into consideration. Most of the previous research does not consider the heterogeneity of the predictability in different regions and uniformly applies the same prediction method, and this can lead to inefficiencies. For example, a Markov-based predictor is four orders of magnitude faster (with only 0.03% computation time) than a computing-intensive NN predictor.

The remainder of this paper is organized as follows. Section II presents how we obtain the maximum predictability  $\Pi^{max}$  and Section III describes the three approaches we implemented for predicting taxi demand. In Section IV, we introduce the data sets we use in this paper and how we preprocess these data. We discuss the results of the taxi demand predictability analysis in Section V. In Section VI, we compare the performance of the three predictive algorithms and show that predictability provides an effective measure for selecting appropriate prediction methods. Related work is discussed in Section VII and we conclude in Section VIII, where we outline directions for future work.

## II. PREDICTABILITY OF TAXI DEMAND

In this section, we formally define maximum predictability  $\Pi^{max}$ . Our goal is to answer the following question with the proposed maximum predictability  $\Pi^{max}$ :

**Problem 1.** *Given a predictive algorithm  $\alpha$  and a sequence of the taxi demand  $D_n^{(i)}$  from time  $1, 2, \dots, n$  at building block  $i$ , considering both the randomness and the temporal correlation of the taxi demand, we want to find the maximum predictability (the highest potential accuracy)  $\Pi^{max}$  that a predictive algorithm  $\alpha$  can reach.*

### A. Taxi Demand

We use the number of taxi pick-ups  $d_t^{(i)}$  to represent the taxi demand at building block  $i$  at time  $t$  ( $1 \leq i \leq m$

Symbol	Meaning
$d_t^{(i)}$	The real taxi demand at the building block $i$ at time $t$ ( $1 \leq t \leq n$ ).
$X_t^{(i)}$	Taxi demand as a random variable.
$D_n^{(i)}$	The historical taxi demand sequence, $d_1^{(i)} d_2^{(i)} d_3^{(i)} \dots d_n^{(i)}$ .
$N^{(i)}$	$N^{(i)}$ is the number of distinct taxi demand at the building block $i$
$S_n^{(i)}$	Time-ordered subsequence $S_n^{(i)}, S_n^{(i)} \subset D_n^{(i)}$
$S_t^{(i)}$	The length of the shortest unseen subsequence starting at the time $t$
$S_{random}^{(i)}$	Random entropy of $D_n^{(i)}$
$S_{Shannon}^{(i)}$	Shannon entropy of $D_n^{(i)}$
$S_{real}^{(i)}$	Real entropy of $D_n^{(i)}$
$\Pi$	The predictability of an arbitrary predictive algorithm $\alpha$
$\Pi^{max}$	The upper bound of predictability $\Pi$ , where $\Pi \leq \Pi^{max}$

TABLE I  
MEANINGS OF MAIN SYMBOLS USED

and  $1 \leq t \leq n$ ). For a given building block  $i$ , we have a sequence of the historical taxi demand as a sequence  $D_n^{(i)} = d_1^{(i)} d_2^{(i)} d_3^{(i)} \dots d_n^{(i)}$ . Note that here  $D_n^{(i)}$  is a sequence representing the taxi demand from time 1 to  $n$ . For example,  $D_n^{(i)} = 2122$  indicates that for block  $i$ , at time step 1 there were 2 pickups, at time step 2 1 pickup, and so on. Table I summarizes the main symbols used in this paper.

Since there can be high variability in taxi demand, it is hard to predict the exact value for  $d_t^{(i)}$ . For simplicity, we use a factor  $q$  to round the value for the taxi demand. We group every  $q$  taxi demand as one taxi demand. We try to predict the taxi demand at every  $q$  level, e.g., if  $q = 10$ , then 620-629 all become 620 in the sequence. After a few tests, we set  $q = 10$  for making the prediction more relaxed while keeping the errors low. We show the effect of  $q$  on  $\Pi^{max}$  in our technical report [14] Appendix D.

### B. Entropy

Entropy is an effective measure to characterize the degree of predictability. In general, a low entropy means higher predictability. We use three entropy measures: the random entropy  $S_{random}^{(i)}$ , the Shannon entropy  $S_{Shannon}^{(i)}$ , and the real entropy  $S_{real}^{(i)}$  [15].

#### 1) Random Entropy:

$$S_{random}^{(i)} = \log_2 N^{(i)} \quad (1)$$

Here  $N^{(i)}$  is the number of unique taxi demand in  $D_n^{(i)}$ , e.g., for the sequence  $D_n^{(i)} = 2122$ , the unique taxi demand  $N^{(i)}$  is 2. In general lower entropy means higher predictability. If the taxi demand in building block  $i$  is low, the random entropy  $S_{random}^{(i)}$  will be a small number and it would be easy to predict taxi demand.

#### 2) Shannon Entropy:

$$S_{Shannon}^{(i)} = - \sum_{t=1}^{N^{(i)}} p(d_t^{(i)}) \log_2 p(d_t^{(i)}) \quad (2)$$

Here  $p(d_t^{(i)})$  is the probability that there is  $d_t^{(i)}$  taxi demand at the building block  $i$  – it characterizes the uncertainty of the taxi demand. Note that both random entropy and Shannon entropy are time-independent and do not consider temporal patterns. A sequence  $D_n^{(i)} = 2122$  has the same random entropy and Shannon entropy as sequence  $D_n^{(i)} = 1222$ .

### 3) Real Entropy:

$$S_{real}^{(i)} = - \sum_{S_n^{(i)} \subset D_n^{(i)}} P(S_n^{(i)}) \log_2 [P(S_n^{(i)})] \quad (3)$$

$P(S_n^{(i)})$  represents the probability of finding a particular time-ordered sub-sequence  $S_n^{(i)}$  in the taxi demand sequence  $D_n^{(i)}$ . Unlike Shannon entropy and random entropy, the real entropy not only considers the frequency of different taxi demand in a building block, but also the order of the temporal patterns of the taxi demand [15].

The problem of finding all the subsets of a given set has exponential complexity ( $O(2^n)$ ). Here we use a Lempel-Ziv estimator to calculate the real entropy. The Lempel-Ziv estimator can rapidly converge to the real entropy [16]. For a taxi demand sequence after time  $n$ , the entropy can be estimated by

$$S_{real}^{(i)} \approx \left( \frac{1}{n} \sum_t s_t^{(i)} \right)^{-1} \ln n \quad (4)$$

Here,  $s_t^{(i)}$  represents the length of the shortest subsequence starting at the time  $t$  which does not appear from 1 to  $t-1$ .

### C. Maximum Predictability $\Pi^{max}$

We define the predictability  $\Pi$  as the success rate that an algorithm  $\alpha$  can correctly predict the future taxi demand at a building block. For a building block with  $N^{(i)}$  unique taxi demand, the predictability measure is subject to the Fano's inequality,  $\Pi \leq \Pi^{max}$  [10]. Given an entropy  $S$  and the distinct taxi demand  $N^{(i)}$  in building block  $i$ , the maximum predictability  $\Pi^{max}$  can be computed by the following equation:

$$S = -\Pi^{max} \log_2(\Pi^{max}) - (1 - \Pi^{max}) \log_2(1 - \Pi^{max}) + (1 - \Pi^{max}) \log_2(N^{(i)} - 1) \quad (5)$$

The maximum predictability  $\Pi^{max}$  is a value between 0 and 1. The larger the value, the more accurate the algorithm is. Due to the space limit, we give a brief proof of the predictability upper bound  $\Pi^{max}$  here. For details, we refer the reader to Appendix E in our technical report [14].

With different entropy measures,  $S_{random}^{(i)}$ ,  $S_{Shannon}^{(i)}$  and  $S_{real}^{(i)}$ , we have different maximum predictability values  $\Pi^{random}$ ,  $\Pi^{Shannon}$  and  $\Pi^{real}$ . It has been proven that  $\Pi^{random} \leq \Pi^{Shannon} \leq \Pi^{real}$  [10], thus  $\Pi^{real}$  is the maximum predictability  $\Pi^{max}$ . Now we can answer the problem proposed in the beginning of the section: given a predictive algorithm  $\alpha$ ,  $\Pi^{max}$  is the maximum predictability that a predictive algorithm  $\alpha$  can reach. The detailed analysis of the maximum predictability of each building blocks in NYC can be found in Section V.

*Proof.* Given a predictive algorithm  $\alpha$ , let  $P_\alpha(X_n^{(i)} = \hat{X}_n^{(i)} | h_{n-1}^{(i)})$  be the distribution generated over the next taxi demand  $\hat{X}_n^{(i)}$  at location  $i$  and  $h_{n-1}^{(i)} = \{X_{n-1}^{(i)}, X_{n-2}^{(i)}, \dots, X_1^{(i)}\}$

the location  $i$ 's past taxi demand from time 1 to  $n - 1$ . Furthermore,  $X_t^{(i)}$  represents the taxi demand at location  $i$  at the time  $t$ , and  $Pr[X_n^{(i)} = x^{(i)} | h_{n-1}^{(i)}]$  is the probability that the next taxi demand  $X_n^{(i)}$  is  $x^{(i)}$  given the taxi demand history  $h_{n-1}^{(i)}$ . Thus  $P(X_n^{(i)} | h_{n-1}^{(i)})$  is the true distribution over the next taxi demand.

Let  $\pi(h_{n-1}^{(i)})$  be the probability that there is a most likely taxi demand at the location  $i$  given the taxi demand history  $h_{n-1}^{(i)}$ . The probability of successfully predicting the next taxi demand is  $Pr_{\alpha}\{X_n^{(i)} = \hat{X}_n^{(i)} | h_{n-1}^{(i)}\}$ . Since  $\pi(h_{n-1}^{(i)}) \geq P(x^{(i)} | h_{n-1}^{(i)})$  for any  $x^{(i)}$ , we have

$$Pr_{\alpha}\{X_n^{(i)} = \hat{X}_n^{(i)} | h_{n-1}^{(i)}\} = \sum_{x^{(i)}} P(x^{(i)} | h_{n-1}^{(i)}) P_{\alpha}(x^{(i)} | h_{n-1}^{(i)}) \quad (6)$$

$$\leq \sum_{x^{(i)}} \pi(h_{n-1}^{(i)}) P_{\alpha}(x^{(i)} | h_{n-1}^{(i)}) \quad (7)$$

$$= \pi(h_{n-1}^{(i)}) \quad (8)$$

Then, we define the predictability  $\Pi(n)$  for a taxi demand sequence with a history of length  $n - 1$ .  $P(h_{n-1}^{(i)})$  represents the probability of a particular taxi demand history  $h_{n-1}^{(i)}$ . If we sum over all the possible histories of length  $n - 1$ , we have the predictability as  $\Pi(n) = \sum_{h_{n-1}^{(i)}} P(h_{n-1}^{(i)}) \pi(h_{n-1}^{(i)})$ .

The overall predictability  $\Pi$  can be defined as  $\Pi = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_t \Pi(t)$ .

Let  $N^{(i)}$  be the total number of possible taxi demand value and there is a uniform distribution over the remaining  $N^{(i)} - 1$  possible taxi demand value. Then we will have  $X'$ , whose distribution  $P'(X^{(i)} | h^{(i)}) = (p, \frac{1-p}{N^{(i)}-1}, \frac{1-p}{N^{(i)}-1}, \dots, \frac{1-p}{N^{(i)}-1})$ . Note here  $S(X_n^{(i)} | h_{n-1}^{(i)}) \leq S(X' | H_{n-1}^{(i)})$ . Then we have  $S(X') = S_F(\pi(h_{n-1}^{(i)}))$ . Here the Fano function  $S_F(p)$  is concave and monotonically decreases with  $p$ . Based on Fano's inequality,  $S(X_n^{(i)} | h_{n-1}^{(i)}) \leq S_F(\pi(h_{n-1}^{(i)}))$ . Following Jensen's inequality, we have

$$S(n) = \sum_{h_{n-1}^{(i)}} P(h_{n-1}^{(i)}) S(X_n^{(i)} | h_{n-1}^{(i)}) \quad (9)$$

$$\leq \sum_{h_{n-1}^{(i)}} P(h_{n-1}^{(i)}) S_F(\pi(h_{n-1}^{(i)})) \quad (10)$$

$$\leq S_F\left(\sum_{h_{n-1}^{(i)}} P(h_{n-1}^{(i)}) \pi(h_{n-1}^{(i)})\right) \quad (11)$$

$$= S_F(\Pi(n)) \quad (12)$$

For a stationary stochastic process  $\chi = X_t^{(i)}$ , based on  $S(n) \leq S_F(\Pi(n))$  and Jensen's inequality, we have

$$S = \lim_{n \rightarrow \infty} \frac{1}{n} S(X_1^{(i)}, X_2^{(i)}, \dots, X_n^{(i)}) \quad (13)$$

$$= \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{t=1}^n S(X_t^{(i)} | h_{t-1}^{(i)}) \quad (14)$$

$$= \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{t=1}^n S(t) \quad (15)$$

$$\leq \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n S_F(\Pi(t)) \quad (16)$$

$$\leq S_F\left(\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n (\Pi(t))\right) \quad (17)$$

$$= S_F(\Pi) \quad (18)$$

Here  $S = \lim_{n \rightarrow \infty} \frac{1}{n} S(X_1^{(i)}, X_2^{(i)}, \dots, X_n^{(i)})$  is the definition of entropy [15]. We define  $S(t) = S(X_t^{(i)} | h_{t-1}^{(i)})$  as the conditional entropy at the time  $t$ . Let  $\Pi^{max}$  be the solution to the equation  $S = S_F(\Pi^{max}) \leq S_F(\Pi)$ . Since  $S_F(p)$  is concave and monotonically decreasing, we have  $\Pi \leq \Pi^{max}$ , which means that  $\Pi^{max}$  is the upper bound of the predictability  $\Pi$ . The predictability upper bound  $\Pi^{max}$  can be solved by the following equation:

$$S = -\Pi^{max} \log_2(\Pi^{max}) - (1 - \Pi^{max}) \log_2(1 - \Pi^{max}) + (1 - \Pi^{max}) \log_2(N^{(i)} - 1) \quad \square$$

#### D. Scalability

The predictability upper bound  $\Pi^{max}$  can be obtained by solving Equation 5. We move  $S$  to the right side of the equation and obtain the following:

$$-\Pi^{max} \log_2(\Pi^{max}) - (1 - \Pi^{max}) \log_2(1 - \Pi^{max}) + (1 - \Pi^{max}) \log_2(N^{(i)} - 1) - S = 0 \quad (19)$$

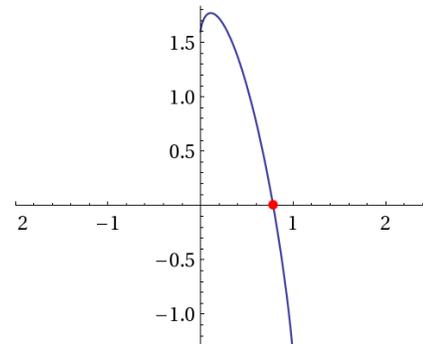


Figure 2. Equation solver for the predictability  $\Pi^{max}$ . Here  $S_{real}^{(i)} = 1.40$ ,  $N^{(i)} = 9$ . By solving the equation  $f(\Pi^{max}) = 0$  we obtain  $\Pi^{max} = 0.78$

In this equation, both  $S$  and  $N^{(i)}$  are known numbers. To solve this equation, we define the function  $f(\Pi^{max}) =$

$-\Pi^{max} \log_2(\Pi^{max}) - (1 - \Pi^{max}) \log_2(1 - \Pi^{max}) + (1 - \Pi^{max}) \log_2(N^{(i)} - 1) - S$ . Since  $\Pi^{max}$  is a value between 0 and 1, finding the intersection when  $f(\Pi^{max}) = 0$  can solve the equation with  $\Pi^{max} = \gamma$ . In Fig. 2, we plot a solution when  $S^{(i)} = 1.40$  and  $N^{(i)} = 9$ ,  $\Pi^{max} = \gamma = 0.78$ . Since both  $S^{(i)}$  and  $N^{(i)}$  are known numbers for a building block  $i$ , the computation time of solving all the equation  $f(\Pi^{max}) = 0$  is  $O(m)$ , where  $m$  is the total number of building blocks. We can scale by distributing the computation of  $f(\Pi^{max}) = 0$  for each building block over multiple processors.

### III. PREDICTORS

We propose three predictors: the Markov predictor (a probability-based method) [12], the LZW predictor (a sequence-based method) [13] and the Neural Network (NN) predictor (a machine learning-based method) [7] for predicting the taxi demand and compare the performance of them. In this section, we answer the following question:

**Problem 2.** *Given the maximum predictability  $\Pi^{max}$  of a building block  $i$ , we want to find the predictor that has a better performance given the trade-off between the computation time and the accuracy.*

#### A. Markov Predictor

We propose the order- $k$   $O(k)$  Markov predictor to predict the future taxi demand from the  $k$  most recent taxi demand sequence  $d_{n-k+1}^{(i)}, d_{n-k+2}^{(i)}, \dots, d_n^{(i)}$ . We define the taxi demand during time  $t$  at the building block  $i$  as a random variable  $X_t^{(i)}$ . Let  $X_{(t,k)}^{(i)}$  donates the sequences of random variable  $X_t^{(i)}, X_{t+1}^{(i)}, X_{t+2}^{(i)}, \dots, X_k^{(i)}$  for  $1 \leq t \leq k \leq n$ . Considering the location  $i$  with a taxi demand history  $D_n^{(i)} = d_1^{(i)} d_2^{(i)} d_3^{(i)} \dots d_n^{(i)}$  and  $N^{(i)}$  as the set of all possible taxi demand at the building block  $i$  as shown in Section II-A, following Markov assumption we have

$$P(X_{n+1}^{(i)} = \beta | X_n^{(i)} = D_n^{(i)}) \quad (20)$$

$$= P(X_{n+1}^{(i)} = \beta | X_{n-k+1,n}^{(i)} = c) \quad (21)$$

$$= P(X_{t+k+1}^{(i)} = \beta | X_{t+1,t+k}^{(i)} = c). \quad (22)$$

Here  $P(X_{n+1}^{(i)} = \beta | X_n^{(i)} = D_n^{(i)})$  means that there are  $\beta$  taxi demand at the building block  $i$  during the time interval  $n + 1$ .  $c$  is the taxi demand where  $d_{n-k+1}^{(i)} d_{n-k+2}^{(i)} \dots d_n^{(i)} = d_{t+1}^{(i)} d_{t+2}^{(i)} \dots d_{t+k}^{(i)} = c$ . We propose a transition probability matrix  $T$  as the Markov predictor (see Alg. 1), the rows and columns in the matrix  $T$  represent the length  $k$  taxi demand sequence. Each element  $T_{c,c'}$  in the matrix represent the prediction  $T_{c,c'} = P(X_{n+1}^{(i)} = \beta | X_n^{(i)} = D_n^{(i)})$ , where  $c = d_{n-k+1}^{(i)} d_{n-k+2}^{(i)} \dots d_n^{(i)}$  and  $c' = d_{n-k+2}^{(i)} d_{n-k+3}^{(i)} \dots d_n^{(i)}$ . The matrix  $T$  provides the immediate probability  $P(X_{n+1}^{(i)} = \beta | X_n^{(i)} = D_n^{(i)})$  of the next symbol  $\beta$  after sequence  $D_n^{(i)}$ .

As  $T$  is unknown, we define an estimate probability  $P'$  from the taxi demand history  $D_n^{(i)}$ , where  $P'(X_{n+1}^{(i)} = \beta | X_n^{(i)} =$

$D_n^{(i)}) = \frac{C(c\beta, D_n^{(i)})}{C(c, D_n^{(i)})}$ . Here  $\frac{C(c\beta, D_n^{(i)})}{C(c, D_n^{(i)})}$  is the probability of the subsequence  $c\beta$  occurs in the sequence  $c$ .

We can estimate the next taxi demand  $X_{n+1}^{(i)}$  at the the building block  $i$  given the order- $k$   $O(k)$  Markov predictor matrix  $T$ . In Alg. 1, we choose the  $\beta$  with the highest probability in the predictor  $T$  that given the recent  $k$  length sequence  $d_{n-k+1}^{(i)} d_{n-k+2}^{(i)} \dots d_n^{(i)} = c$ . Note that the Markov predictor  $T$  might return an empty value given the recent taxi demand sequence  $c$ . This is due to the fact that the taxi demand history  $d_{n-k+1}^{(i)} d_{n-k+2}^{(i)} \dots d_n^{(i)} = c$  did not occur previously. In this case, we return the taxi demand with highest probability in  $D_n^{(i)}$ . We set  $k = 3$  here to improve the prediction accuracy while reducing the computation time [17].

It is easy to maintain and update the Markov predictor matrix  $T$  in Alg. 1. For making the prediction, the predictor can scan the matrix  $T$  at the row  $c$  and pick the next taxi demand with the highest probability. Then the order- $k$   $O(k)$  Markov predictor updates the row of  $c$  with the new taxi demand in the matrix  $T$  after time  $n + 1$ . Note that here  $T$  is a  $c' \times N^{(i)}$  matrix,  $c'$  is the unique taxi demand subsequence and  $N^{(i)}$  is the unique taxi demand in  $D_n^{(i)}$ .

---

#### Algorithm 1: Order $O(k)$ Markov Predictor

---

**input** : The taxi demand sequence  $D_n^{(i)}$ , the recent sequence length  $k$ , the time  $n$  and the Markov Predictor matrix  $T$   
**output**: The predicted taxi demand  $\beta$  at the time  $n + 1$   
1 Extracting the recent length  $k$  subsequence  $c = d_{n-k+1}^{(i)} d_{n-k+2}^{(i)} \dots d_n^{(i)}$  from  $D_n^{(i)}$  ending at the  $n$ ;  
2 Finding the subsequence  $c$  subsequence in  $T$  at the row  $T(c)$ ;  
3 **if**  $T(c)$  is empty // The taxi demand sequence  $c$  did not appear previously.  
4 **then**  
5     return the taxi demand with the highest probability in  $D_n^{(i)}$ ;  
6 **else**  
7     Finding the  $c\beta$  with the highest probability at the row  $T(c)$ ;  
8     // Similar taxi demand history has appeared at least once before  
9     return  $\beta$ ;  
10 Update the matrix  $T$  with the new taxi demand  $d_{n+1}^{(i)}$  at the time  $n + 1$ .

---

#### B. Lempel-Ziv-Welch (LZW) predictor

The LZW predictor is based on the Lempel-Ziv-Welch text encoding algorithm (LZW algorithm). Given a taxi demand sequence  $D_n^{(i)}$ , LZW algorithm partitions  $D_n^{(i)}$  into distinct subsequence  $s_0^{(i)}, s_1^{(i)}, s_2^{(i)}, s_3^{(i)}, \dots, s_m^{(i)}$ , where  $s_t^{(i)}$  represents the shortest subsequence starting at the time  $t$  which does not appear from 1 to  $t - 1$  (see Alg. 2). For example, in Appendix [14] Supplementary Figure S1, the taxi demand history  $D_n^{(i)} = 112112132$  can be parsed into subsequence 1, 2, 3, 11, 12, 21, 121, 13, 32 by LZW algorithm.

We build a LZW tree to maintain the LZW predictor (see Alg. 2). The LZW tree is growing dynamically while parsing the taxi demand history  $D_n^{(i)}$ . The root of the tree is a empty list. Each node represents one taxi demand subsequence  $s_t^{(i)}$  starting from the root with the sequences of nodes encountered on the path to the node. Besides the taxi demand  $d_t^{(i)}$ , each node also maintains a counter  $count_{d_t^{(i)}}$  to calculate

the occurrences of the subsequence  $d_t^{(i)}$ . We define the taxi demand during time  $t$  at the building block  $i$  as a random variable  $X_t^{(i)}$ . We have the LZW predictor as follows:

$$P(X_{n+1}^{(i)} = \beta | D_n^{(i)}) = \frac{NLZ(s_m^{(i)}\beta, D_n^{(i)})}{NLZ(s_m^{(i)}, D_n^{(i)})} \quad (23)$$

Similar to Markov predictor, here  $\frac{NLZ(s_m^{(i)}\beta, D_n^{(i)})}{NLZ(s_m^{(i)}, D_n^{(i)})}$  represents the probability that the subsequence  $s_m^{(i)}\beta$  occurs in the sequence  $s_m^{(i)}$ . While the Markov predictor considers how often does the sequence of interest occur in the entire taxi demand sequence  $D_n^{(i)}$ , the LZW predictor, only considers the sequence of interest in the partitions  $s_t^{(i)}$ .

---

### Algorithm 2: LZW Predictor

---

**input** : The taxi demand sequence  $D_n^{(i)}$ , the current time  $n$   
**output**: The predicted taxi demand  $\beta$  at the time  $n + 1$   
1 Extracting all the taxi demand and put them in the table;  
2 Initializing sequence  $g$  to be the first taxi demand  $d_t^{(i)}$  in  $D_n^{(i)}$ ;  
3 Building an empty LZW Tree;  
4 **while** Any taxi demand input left in  $D_n^{(i)}$ ;  
5 **do**  
6      $g$  is the next taxi demand after  $c$ ;  
7     **if**  $g + h$  is in the tree;  
8         **then**  
9              $g = g + h$ ;  
10         **else**  
11             Update the LZW tree with  $g + h$ ;  
12             Update  $count_{d_t^{(i)}}$  at each node;  
13              $g = h$ ;  
14 Finding the  $s_m^{(i)}\beta$  with the highest probability in the the LZW tree;  
15 return  $\beta$ ;

---

### C. Neural Network Predictor

We propose a multilayer perception Neural Network (NN) predictor to predict the next taxi demand (see Alg. 3). The neural network predictor we implemented consists of two layers: Sigmoid and Softmax (see Appendix [14] Supplementary Figure S2). We adopt the same input data, i.e., "temperature", "precipitation", "wind speed", "day of the week" and "hour of the day", as previous research for predicting taxi demand [7]. Both layers contain 100 neurons. Layers are based on the standard sigmoid and softmax functions [18], and the output for each of the layers is calculated as:

$$S(\sum_i w_i d_t^{(i)} + b). \quad (24)$$

Here,  $w_i$  represents the weights and  $d_t^{(i)}$  represents input demand and  $b$  represents bias. When the neural network is trained, the prediction results are linearly convoluted using the Gaussian kernel with the nearest demand time-wise:

$$G(\hat{D}_{n+1}^{(i)}) = \sum_t \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{D_{n+1}^{(i)} - D_t^{(i)}}{2\sigma^2}} \quad (25)$$

where  $D_{n+1}^{(i)}$  ( $\beta$ ) stands for a predicted demand at the time  $n + 1$ ,  $D_{n-t}^{(i)}$  stands for the  $t^{th}$  nearest known demand.

---

### Algorithm 3: NN Predictor

---

**input** : The taxi demand sequence  $D_n^{(i)}$  from time stamps  $T = \{1, \dots, n\}$  per building block  
**output**: The predicted taxi demand  $\beta$  at time stamps  $n + 1$   
1 Creating multiple features from the timestamps: each of the time periods and dates is assigned an individual binary feature that turns on during the period and date. Creating derivative features for weekdays;  
2 Training a separate MLP neural network for each building block.;  
3 Calculating Gaussian convolution of the demand  $\hat{D}_{n+1}^{(i)}$  predicted by the neural network and the time-wise nearest values  $\{D_{n-1}^{(i)}, D_{n-2}^{(i)}, D_{n-3}^{(i)}, \dots\}$ ;  
4 return  $\hat{D}_{n+1}^{(i)}$  as  $\beta$ ;

---

## IV. DATA SETS AND PREPROCESSING

In this section, we introduce the data sets we used in this paper and how we preprocess the data. We use the NYC yellow taxi data set, the NYC Uber taxi data set, and NYC Pluto data set. These data sets provide a good coverage of the city with respect to both space and time.

### A. Data Sets

**Taxi data set.** The New York yellow taxi data set is a public data set provided by the Taxi and Limousine Commission (TLC) [19]. It includes trip records from all trips completed in yellow taxis in New York City in June, 2014. Each trip record contains pick-up and drop-off time, pick-up and drop-off locations, trip distances, itemized fares, and driver-reported passenger counts. The data was recorded through meters installed in each taxi. In total we have 13,813,031 taxi pick-up records from 13,237 yellow taxis.

The New York Uber data set is a public data set provided by the TLC [19]. This data set contains 663,845 Uber pick-up records in June 2014. The key information provided by these data sets is summarized in our technical report [14] Supplementary Table S1. We extract the following information from the data set: taxi id, pickup time and the corresponding pick-up location (building block).

**Pluto data set.** We use the Pluto (Primary Land Use Tax Lot Output) data set to map the GPS points with the associated building blocks and provide the land use information for each building block [20]. The Pluto data set is an extensive land use and geographic data set provided by NYC Department of City Planning. It contains detailed information about every piece of land in the city, including year built, number of units, and lot size.

### B. Data Preprocessing.

We use the Pluto building shape file to map the pick-up GPS points with the associated building blocks: if a building block is within 200 ft radius of the pick-up location, we consider that building as the one passengers getting on the taxi and there is one taxi demand at that building block. If multiple building blocks are within 200 ft radius of the pick-up location, we consider the nearest building block as the one that has a taxi demand. There are over 43,000 building blocks in Manhattan, however, for meaningful analyses, only those that have sufficient temporal coverage were included in our study. We found 9,940 such building blocks in our data set

with at least 5 pick-ups a day. For the rest of the paper, we only use these 9,940 building blocks, unless otherwise noted. After the map matching process, we obtain the taxi demand sequence for each building block (Building block id  $i$ , taxi demand sequence  $D_n^{(i)} = d_1^{(i)} d_2^{(i)} d_3^{(i)} \dots d_n^{(i)}$ ).

**Scalability.** All of our data preprocessing were conducted using the operational data facility at our research center. In particular, the mapping of taxi pickups to geospatial features, which requires a lot of processing given the volume of the pickup trips, on a 1200+ core cluster running Cloudera Data Hub 5.4 with Apache Spark 1.6. The cluster consists of 20 high-end nodes, each with 24TB of disk, 256GB of RAM, and 64 AMD cores. It takes about ten minutes for our R-tree based [21] algorithm to map matching the 14 million samples.

## V. RESULTS

In this section, we analyse the predictability of taxi demand over each building block in NYC. We show that the maximum predictability of the taxi demand can reach up to 83% accuracy in average.

### A. Limits of Predictability

We show the distribution of the entropy and the maximum predictability obtained from the yellow taxi data set in Fig. 3. Here we group taxi demand every hour and study the hourly taxi demand. We first determine the entropy  $S$  and maximum predictability  $\Pi$  of the taxi demand of each building block using the yellow taxi data set. Then we obtain the distribution of the entropy  $S$  and maximum predictability  $\Pi$  over all building blocks. Fig 3(a) depicts the entropy distribution of real Entropy  $S_{random}^{(i)}$ , Shannon Entropy  $S_{Shannon}^{(i)}$  and random Entropy  $S_{real}^{(i)}$ ; whereas Fig 3(b) presents the distribution of  $\Pi^{max}$ ,  $\Pi^{Shannon}$ ,  $\Pi^{random}$  respectively.

We find that the distribution of  $S_{random}^{(i)}$  peak at  $S_{random}^{(i)} = 3.6$  (see Fig. 3 (a)). It implies that a building block would have  $N^{(i)} = 2^{3.6} \approx 12$  distinct taxi demand levels. That is, almost every two hours we will observe a new taxi demand level compared to the previous one. Recall that we define  $q = 10$  when categorizing the taxi demand, which implies that there are about 10 taxi demand differences every two hour in the same building block. In contrast, the real entropy  $S_{real}^{(i)}$  peaks at a much smaller entropy value,  $S_{real}^{(i)} = 0.9$ . As discussed in Section. II-B, the real entropy captures the temporal correlation of the taxi demand sequence. The small real entropy  $S_{real}^{(i)}$  means a high temporal pattern.

The predictability that any algorithm can correctly predict the next taxi demand is  $\Pi$ , and the upper bound is  $\Pi^{max}$ . From the distribution of the maximum predictability  $\Pi^{max}$  shown in Fig. 3 (b), we observe that average value of  $\Pi^{max}$  is 0.83. It indicates that the taxi demand can be potentially correctly predicted with an accuracy 83% over all the building blocks. Both  $\Pi^{random}$  and  $\Pi^{Shannon}$  are smaller than  $\Pi^{max}$ , which is constant with the previous findings [10],  $\Pi^{random} \leq \Pi^{Shannon} \leq \Pi^{max}$ . Since  $\Pi^{max}$  captures the temporal correlation of taxi demand, we can reach a higher

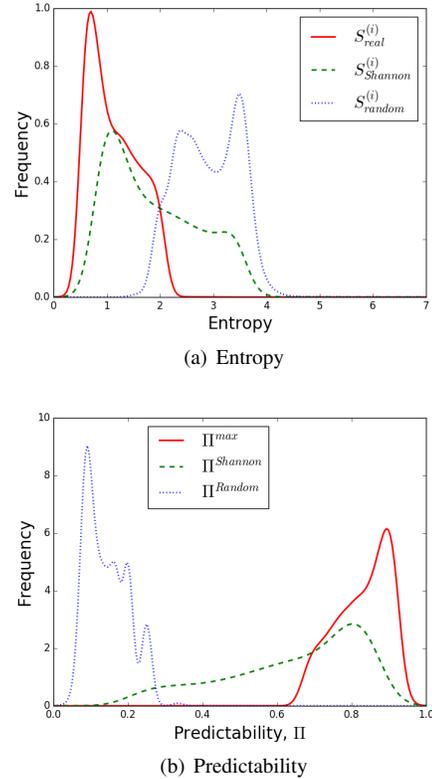


Figure 3. Distribution of (a) Entropy (b) Predictability of the taxi demand over all building blocks. The red line, the green line and the blue line refer to the probability density function of  $S_{random}^{(i)}$ ,  $S_{Shannon}^{(i)}$ ,  $S_{real}^{(i)}$  and  $\Pi^{max}$ ,  $\Pi^{Shannon}$ ,  $\Pi^{random}$  in (a) and (b).

potential predictive accuracy if considering the temporal correlation in our predictive algorithms. Similar results can also be found if we group the taxi demand daily instead of hourly (see our technical report [14] Supplementary Fig. S3).

### B. Predictability of Different Functional Buildings

In Fig. 4 we plot the heat map of hourly taxi demand predictability in Manhattan. From the figure we observe that different building blocks exhibit different maximum predictability. In the working places such as Lower Manhattan or residential places such as East Village, they have higher predictability (temporal correlation). People usually go work in the morning and come home during night, thus the predictability in these areas are higher compared to other places.

To further evaluate the predictability of different functional regions, in Table II we show the predictability of taxi demand of building blocks with different land use. Both residential and working places have high predictability and thus exhibit strong temporal patterns. For other places such as hotels, transportation centers or bureau properties, the predictability is not as high as the residential or working places. The taxi demand for these regions is mainly dependent on the events happened during the day. E.g., a boat arrives at the pier or guests check out in a hotel.

Building Classes: First Level	Building Classes: Second Level	Maximum Predictability $\Pi^{max}$
H. HOTELS	H1. Luxury Type, Built Prior to 1960 & Over	0.69
L. LOFTS	L2. Fireproof Loft & Storage & Over	0.71
O. OFFICE BUILDINGS	O3. Ten Stories & Over	0.72
	O4. Tower Type	0.70
R. CONDOMINIUMS	RC. Mixed Commercial/Condos	0.71
	RK. Store Buildings, Retails	<b>0.72</b>
	RH. Hotel/Boatel	0.71
S. RESIDENCE, MULTIPLE USE	S3. 3-Family With Store/Office	0.71
T. TRANSPORTATION FACILITIES	T2. Piers, Docks, Bulkheads	0.69
U. UTILITY BUREAU PROPERTIES	U6. Railroads, Private Ownership	<u>0.67</u>

TABLE II  
PREDICTABILITY OF DIFFERENT LAND USES



Figure 4. Heat map of predictability in Manhattan

### C. Uber Versus Yellow Taxi

In this section we examine the difference of the predictability between the Uber and yellow taxis. Due to the sparsity of the Uber taxis, we analyse the maximum predictability of both data sets at neighborhood level. We group the hourly taxi demand in each neighborhood and examine the maximum predictability of the hourly taxi demand sequence. As shown in Fig. 5, the maximum predictability of the Uber taxi service is higher than the yellow taxis. This is possibly because that the yellow taxis usually use a random cruising strategy, while the Uber taxis go to the passenger's places when a request is received. The temporal correlation of the taxi demand in a region can be better captured by the Uber taxi. Similar results can also be found in our technical report [14] Supplementary Fig. S4 with daily taxi demand sequence.

## VI. EVALUATION

In this section, we evaluate three predictors, the Markov predictor [12], the LZW predictor [13], the NN predictor [7] and examine which prediction algorithm can approach the prediction upper bound  $\Pi^{max}$ .

### A. Experiment Setup

We conduct our experiments, which consists of three predictors, on a workstation with dual Intel Xeon E5-2695 2.4GHz

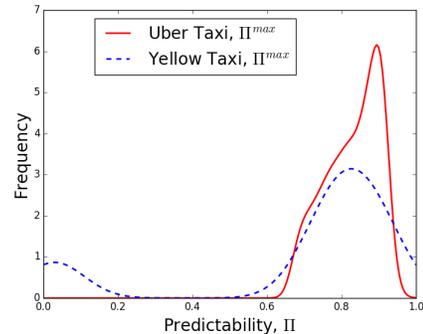


Figure 5. Distribution of  $\Pi^{max}$  of the Uber and yellow taxis

processors (12 cores in total) and 32 GB of memory. Our algorithms were implemented in Python and also made use of the scikit-learn package [22]. Inspired by previous work [4], we use three weeks taxi pick-up data as the training data, and one week taxi pick-up data as test data for comparing the three predictors. We group the taxi demand hourly for each building block. To reduce the bias, we pick 12 time intervals, Monday, Wednesday and Sunday at midnight (00:00), morning (06:00), noon (12:00) and evening(18:00) in the last week as the prediction target. We rank the building blocks by the maximum predictability  $\Pi^{max}$ , and group them into ten 1,000 building block groups. We pick up the a set of median building blocks of each group to investigate (10 in each group). In total we have 60,480 training samples and 1,200 testing samples.

**Error metrics.** We employ an error measurement value, symmetric Mean Absolute Percentage Error (sMAPE [23]), to evaluate the prediction algorithms. For a given building block  $i$ , let  $R_n^{(i)} = r_1^{(i)} r_2^{(i)} r_3^{(i)} \dots r_n^{(i)}$  donates the predicted the taxi demand sequence. We have the real taxi demand sequence  $D_n^{(i)} = d_1^{(i)} d_2^{(i)} d_3^{(i)} \dots d_n^{(i)}$ .  $sMAPE^{(i)}$  is defined as follows (for more details, please refer to our technical report [14] Appendix F.):

$$sMAPE^{(i)} = \frac{1}{n} \sum_{t=1}^n \frac{|d_t^{(i)} - r_t^{(i)}|}{d_t^{(i)} + r_t^{(i)} + c} \quad (26)$$

### B. Evaluation

We show the sMAPE error of the three predictors and the prediction error bound  $1 - \Pi^{max}$  in Fig. 6 (a). We have ten

building block groups with  $\Pi^{max}$  increasing from low to high (E.g., 0%-10% means the bottom 10% building block group according to  $\Pi^{max}$ ). For each group we calculate the average prediction error. We find that the NN predictor provides better accuracy for building blocks with low predictability. In the group with lowest maximum predictability  $\Pi^{max} = 72\%$ , the NN predictor has a prediction accuracy 70%-the highest of all predictors. This is because that the NN predictor is able to capture the multiple features such as the weather information [7], which can not be captured by other two predictors.

The Markov predictor provides better accuracy for building blocks with high predictability, and converges to the predictability upper bound  $\Pi^{max}$  quickly. This is constant with previous research that there is a high positive correlation between the maximum predictability and the Markov predictor prediction accuracy [12]. In the building blocks with high maximum predictability, the Markov predictor is able to predict the taxi demand with an 89% accuracy, 11% better than the NN predictor. Besides, the computation time of the Markov predictor is only 0.5 seconds, 0.03% of the NN predictor (see Fig. 6 (b)). The Markov predictor is able to provide better prediction accuracy with much less computation time for the areas with high predictability.

From the experiment we find that the maximum predictability  $\Pi^{max}$  can help us determining which predictor to use. In the areas with low predictability we can use the NN predictor to reach high accuracy by capturing the multiple features, while in the areas with high predictability the Markov predictor is able to provide better prediction accuracy while keeping the computation time low.

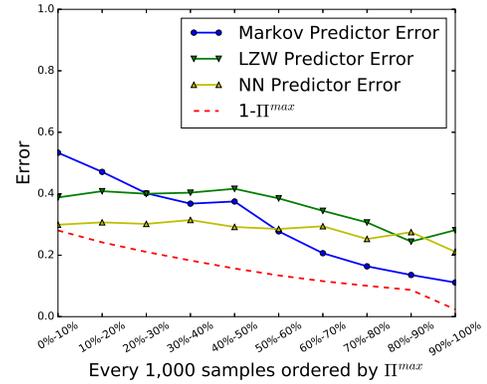
## VII. RELATED WORKS

### A. Predicting Taxi Demand

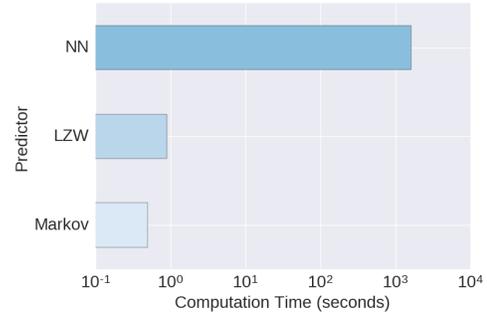
Taxi demand prediction problem has attracted more attentions recently due to the available of taxi data set. Mukai et. al forecast the taxi demand from taxi historical data with a neural network (multilayer perceptron) [7]. Li Et. al adapt the feature selection tool, L1-Norm SVM, to select the most salient feature patterns that determining the taxi performance [3]. An improved ARIMA based prediction method to forecast the spatial-temporal variation of passengers in hotspots is proposed by Li et. al [5]. Moreira-Matias et. al argue that ARIMA prediction method is not the best solution [4]. They propose a new ensemble framework and show that their ensemble work can reach a high prediction accuracy.

### B. Inferring Unmet Demand

From the taxi data set we can measure and predict the met taxi demand, that is, the number of the taxi services emerged and will emerge at different locations. However, the unmet taxi demand, e.g., the number of people who need a taxi but could not find one, can not be simply extracted from the taxi data set. To solve this problem, recent papers try to infer the unmet taxi demand from the taxi data set. In [9] the authors combine flight arrival with taxi demand and predict the passenger demand at different airport terminals in Singapore use queueing theory.



(a) Prediction Errors



(b) Computation Time

Figure 6. (a) Prediction error and (b) Computation time of the Markov, LZW and NN predictor.

Anwar et. al [8] formalize the unmet taxi demand problem and present a novel heuristic algorithm to estimate it without any additional information. They infer the unmet taxi demand from taxis with empty services and show that it can be used to quantity the unmet demand. It must be noted that, although in our paper we only focus on predicting the met taxi demand of different building blocks. Our method is a general solution and can be used for predicting unmet taxi demand.

### C. Temporal Pattern of Human Mobility

It has been found that urban human mobility exhibit strong regularities, e.g., people usually go to work during daytime on weekdays, and go home after work. Marta et al. find that the trajectories in human mobility exhibit strong regularities by studying cell phone user's locations [24]. They show that human trajectories exhibit a high degree of temporal and spatial temporal pattern. Each person has a significant probability to return to a few highly frequented locations such as home or working places. Song et al. propose the entropy-based probability to measure the temporal pattern of the individual human mobility [10]. They find an potential prediction algorithm can reach up to 93% accuracy. They also observe that the user can be found in his or her most visited location during the corresponding hour long period with a high probability, which also indicates the high temporal pattern of human mobility. The temporal pattern of human mobility also leads to the temporal pattern of taxi pick-ups. The human

mobility patterns for different functional regions are different. Previous papers analyse the temporal pattern of urban human mobility and infer the functions of the regions in three cities [25], [26]. Similar results can also be found in Table II, where there is a high predictability in the residential places and a low predictability in the transportation hubs.

### VIII. CONCLUSION

In this paper, we analyse over 14 million yellow and Uber taxi pick-up samples in NYC. We find that there is a high predictability of taxi demand (up to 83% in average), which indicates strong temporal correlation of human mobility. We also examine which predictive algorithm can approach the maximum predictability. We show that the compute-intensive NN predictor does not always have better prediction accuracy than the Markov one. In the areas with low predictability, the NN predictor can reach high accuracy by capturing additional features (weather, etc). On the other hand, in the areas with high predictability, the Markov predictor is able to reach high prediction accuracy while keeping the computation time low. We also find that, the temporal correlation of the taxi demand can be better captured in the Uber taxi data, possibly due to different cruising strategies.

It must be noted that our results is not limited to taxi demand problem. Same approach can be used for predictions in other spatio-temporal data sets, such as mobile traffic of a cellular tower [27], the number of tweets at a location [28] or the bike usage of a bike station [29]. In the future work, we will propose a new predictability-based predictor for the spatio-temporal prediction problem, e.g., the output of the Markov model can be viewed as an feature for NN predictor.

### ACKNOWLEDGMENT

The authors thank: the New York City TLC for providing the data used in this paper; Harish Doraiswamy for his feedback on an early draft of this paper; and the anonymous reviewers for their insightful comments and suggestions. This work was supported in part by a CUNY IRG Award, the NYU Center for Urban Science and Progress, NSF awards CNS-1229185, CI-EN-1405927 and CNS-1544753, and by the Moore-Sloan Data Science Environment at NYU. Juliana Freire is partially supported by the DARPA Memex program.

### REFERENCES

- [1] Y. Huang and J. W. Powell, "Detecting regions of disequilibrium in taxi services under uncertainty," in *SIGSPATIAL'12, Redondo Beach, CA, USA, November 7-9, 2012*, 2012, pp. 139–148.
- [2] F. Miao, S. Han, S. Lin, Q. Wang, J. A. Stankovic, A. Hendawi, D. Zhang, T. He, and G. J. Pappas, "Data-driven robust taxi dispatch under demand uncertainties," *CoRR*, vol. abs/1603.06263, 2016.
- [3] 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 *IEEE PerCom 2011, 21-25 March 2011, Seattle, WA, USA, Workshop Proceedings*, 2011, pp. 63–68.
- [4] L. Moreira-Matias, J. Gama, M. Ferreira, J. Mendes-Moreira, and L. Damas, "Predicting taxi-passenger demand using streaming data," *IEEE Trans. Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1393–1402, 2013.
- [5] X. Li, G. Pan, Z. Wu, G. Qi, S. Li, D. Zhang, W. Zhang, and Z. Wang, "Prediction of urban human mobility using large-scale taxi traces and its applications," *Frontiers of Computer Science in China*, vol. 6, no. 1, pp. 111–121, 2012.
- [6] J. Yuan, Y. Zheng, L. Zhang, X. Xie, and G. Sun, "Where to find my next passenger," in *UbiComp 2011, Beijing, China*, 2011, pp. 109–118.
- [7] N. Mukai and N. Yoden, *Taxi Demand Forecasting Based on Taxi Probe Data by Neural Network*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 589–597.
- [8] A. Afian, A. Odoni, and D. Rus, "Inferring unmet demand from taxi probe data," in *ITSC 2013*, Sept 2013, pp. 861–868.
- [9] A. Anwar, M. Volkov, and D. Rus, "Changinow: A mobile application for efficient taxi allocation at airports," in *ITSC 2013*, Oct 2013, pp. 694–701.
- [10] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, "Limits of predictability in human mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010.
- [11] C. Zhong, E. Manley, S. M. Arisona, M. Batty, and G. Schmitt, "Measuring variability of mobility patterns from multiday smart-card data," *J. Comput. Science*, vol. 9, pp. 125–130, 2015.
- [12] X. Lu, E. Wetter, N. Bharti, A. J. Tatem, and L. Bengtsson, "Approaching the Limit of Predictability in Human Mobility," *Scientific Reports*, vol. 3, Oct. 2013.
- [13] C. T. Cheng, R. Jain, and E. van den Berg, "Mobile wireless systems: Location prediction algorithms," in *Encyclopedia of Wireless and Mobile Communications*, 2008.
- [14] K. Zhao, D. Khryashchev, J. Freire, C. Silva, and H. Vo, "Predicting taxi demand at high spatial resolution: Approaching the limit of predictability," <https://serv.cusp.nyu.edu/projects/demandprediction/2016taxidemand.pdf>, Tech. Rep., July 2016.
- [15] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley-Interscience, 2006.
- [16] I. Kontoyiannis, P. H. Algoet, Y. M. Suhov, and A. J. Wyner, "Non-parametric entropy estimation for stationary processes and random fields, with applications to English text," *Information Theory, IEEE Transactions on*, vol. 44, no. 3, pp. 1319–1327, 1998.
- [17] L. Song, D. Kotz, R. Jain, and X. He, "Evaluating location predictors with extensive wi-fi mobility data," in *Proceedings IEEE INFOCOM 2004, Hong Kong, China, March 7-11, 2004*, 2004.
- [18] E. Alpaydin, *Introduction to Machine Learning*, 2nd ed. The MIT Press, 2010.
- [19] "New York Taxi data set," <http://www.nyc.gov/html/tlc>.
- [20] "New York Pluto data set," <http://www1.nyc.gov/site/planning/data-maps/open-data.page>.
- [21] A. Guttman, "R-trees: A dynamic index structure for spatial searching," in *SIGMOD'84*, 1984, pp. 47–57.
- [22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [23] S. Makridakis and M. Hibon, "The m3-competition: results, conclusions and implications," *International Journal of Forecasting*, vol. 16, no. 4, pp. 451–476, 00 2000.
- [24] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility patterns," *Nature*, vol. 453, no. 7196, pp. 779–782, June 2008.
- [25] K. Zhao, M. P. Chinnasamy, and S. Tarkoma, "Automatic city region analysis for urban routing," in *IEEE ICDMW 2015, Atlantic City, NJ, USA, November 14-17, 2015*, 2015, pp. 1136–1142.
- [26] J. Yuan, Y. Zheng, and X. Xie, "Discovering regions of different functions in a city using human mobility and pois," in *KDD'12*, 2012, pp. 186–194.
- [27] H. Wang, F. Xu, Y. Li, P. Zhang, and D. Jin, "Understanding mobile traffic patterns of large scale cellular towers in urban environment," in *Proceedings of the 2015 ACM Internet Measurement Conference, IMC 2015, Tokyo, Japan, October 28-30, 2015*, 2015, pp. 225–238.
- [28] T. Cheng and T. Wicks, "Event detection using twitter: A spatio-temporal approach," *PLoS ONE*, vol. 9, pp. 1–10, 06 2014.
- [29] D. Singhvi, S. Singhvi, P. Frazier, S. Henderson, E. Mahony, D. Shmoys, and D. Woodard, "Predicting bike usage for new york citys bike sharing system," 2015.