A Probabilistic Approach to User Mobility Prediction for Wireless Services

David Stynes, Kenneth N. Brown, Cormac J. Sreenan
CTVR, Dept. of Computer Science,
University College Cork, Ireland

***Abstract*—Mobile and wireless networks have long exploited mobility predictions, focused on predicting the future location of given users, to perform more efficient network resource management. In this paper, we present a new approach in which we provide predictions as a probability distribution of the likelihood of moving to a set of future locations. This approach provides wireless services a greater amount of knowledge and enables them to perform more effectively. We present a framework for the evaluation of this new type of predictor, and develop 2 new predictors, HEM and G-Stat. We evaluate our predictors accuracy in predicting future cells for mobile users, using two large geolocation data sets, from MDC [11], [12] and Crawdad [13]. We show that our predictors can successfully predict with as low as an average 2.2% inaccuracy in certain scenarios.**

***Index Terms*—Mobility Prediction, Mobile networking, Mobility and Nomadicity, Location Based Services**

I. Introduction

Mobility predictions have regularly been shown to be a necessity for providing efficient resource management and services in wireless networks: foreknowledge of users' mobility allows for more efficient handover management [1], reducing the amount of signalling and interruption time. Content pre-fetching [2] relies upon predictions to know which locations should pre-fetch data, in order to improve performance and energy efficiency of the mobile system. Opportunistic caching [3] for handovers in a mobile system that utilises a passive optical network backhaul relies upon predictions to efficiently use the restricted memory space available at base stations for caching to improve handovers. Future mobile technologies are shifting toward smaller cell sizes, such as Femtocells, to improve spectrum re-use and mobility predictions are necessary to decrease the amount of unnecessary handovers in these dense small cell topologies [4]. Many location-based services [5], such as shared ride recommendations or targeted ads, are also heavily dependant upon predictions to provide a good quality of service.

A wide range of approaches for providing mobility predictions, including Markov-based [1], Compression-based [6], Mixture model-based [7], Trajectory-based [8] and many others have been proposed, all with the singular aim of providing a predicted future location or locations, either in the short term or the long term, for a given mobile user. However, this format of predictions is too restrictive. By providing predictions of only the most likely future location for a user, we are depriving services of a great deal of useful information that could positively influence their behaviour. If instead, services had full knowledge of the probabilities of moving to each possible future location, they could make more informed decisions and provide a more efficient utilisation of resources.

Taking as an example, services like content pre-fetching [2] and opportunistically caching for handovers [3], in which base stations, or access points, providing a service have limited resources available which must be distributed between users who may handover soon. The resources can be more effectively allocated between users predicted to handover when given full knowledge of how likely it is that each will handover to a location. This is not possible when the users are indistinguishable due to only the most likely future location being provided, as in the current format of predictions. Full knowledge also means that base stations which are not the most probable future location have the option to allocate resources for possible handovers if they choose. Other services that seek to influence the mobility of users will also benefit from knowing these unlikely destinations. For a ride-sharing system, such as Uber [9] or Lyft [10], knowing there is a low probability of drivers travelling to a location can allow it to offer bonus credit to influence drivers to go by the unlikely route, if they predict a high demand for rides in that region. Conversely, if there is a high chance of drivers going to the location naturally, they would not wish to offer a bonus.

In this paper, we propose a new scheme for the provision of mobility predictions in a probability distribution format (PDF), specifying the probability of the user moving to each possible future location. These PDF-predictions allow a greater degree of control and reasoning for services that utilise predictions. While the underlying techniques of many traditional predictors are also in PDF, their analysis has been limited to measuring accuracy of the highest valued probability (i.e. most likely future location). We discuss the key differences in evaluating PDF-predictions versus traditional predictions and provide a framework for evaluating all probabilities. We present two new methods of providing PDF-predictions: The first method, History-based Expectation-Maximisation (HEM), utilises an Expectation-Maximisation (EM) algorithm to generate a series of DTMCs that provide predictions taking account of users' mobility history. Our second method, G-Stat, augments HEM to provide better predictions that incorporate GPS trajectory-based predictions. We compare both against a standard Markov predictor for traditional predictions adapted to provide PDF-predictions and evaluate the performance of our methods on two large geolocation data sets from the Mobile Data

978-1-5090-0304-4/16/$31.00 ©2016 IEEE

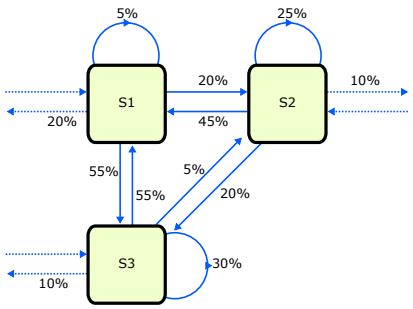

Fig. 1. A sample view of a portion of a DTMC.

Challenge (MDC) [11], [12] and Crawdad's Rome Taxis [13]. We show that G-Stat can achieve as low as 2.2% average inaccuracy on certain data.

II. Two Methods of Mobility Prediction

We now present two methods of providing PDF-mobility predictions through a dense deployment of small cells. We assume that every location in the region of mobility is associated with one primary cell from the set of cells, $S = \{s_1, s_2, ..., s_o\}$. For simplicity when evaluating the algorithms in Section IV we divide the area into a grid of rectangular cells, but both of these methods are compatible with any shapes and topologies of cells. Both methods require training data to be supplied in order to learn movement patterns before they can be used. The training data takes the form of sequences of the connected-cell transitions that occurred at fixed time intervals for a set of m users, $U = \{u_1, u_2, ..., u_m\}$.

A. A Simple Discrete Time Markov Chain-Based Predictor (sDTMC)

The first method, a single Discrete Time Markov Chain (sDTMC), is a well known type of Markov chain predictor [6] that has been used as the core of many traditional mobility prediction techniques in mobile wireless systems, which we use here as a PDF-predictor for our analysis. Predictions take the form of a probability distribution, $P(s_i \rightarrow s_j)$, for transitioning from cell s_i to cell s_j in the next time interval, where $\sum_{s_j \in S} P(s_i \rightarrow s_j) = 1, \forall s_i \in S$. Figure 1 shows a representation of a sample DTMC. The history of cells is not relevant for the prediction and only the current cell, s_c, is used to determine the probability of heading to each possible future cell, s_f. For example, from the figure there is a 20% chance of transitioning to cell s_2 from s_1, $P(s_1 \rightarrow s_2) = 0.20$, and a 55% chance of transitioning to s_3 from s_1, $P(s_1 \rightarrow s_3) = 0.55$, and neither is dependent on where the user was prior to entering cell s_1. We note that it is also possible for a transition from a cell to the same cell, i.e. to remain in the same cell over a time interval. We define the values of the cell transition probabilities as:

$$P(s_c \rightarrow s_f) = \frac{N(s_c \rightarrow s_f)}{\sum_{s_j \in S} N(s_c \rightarrow s_j)}, \forall s_c, s_f \in S$$

where $N(s_i -> s_j)$ is the number of transitions from cell s_i to cell s_j in the entire training data set. This DTMC therefore provides probability distributions that predict that the chances of transitioning to cell s_f from cell s_c is equal to the proportion of all transitions from cell s_c which were transitions to cell s_f, in the training data set. sDTMC is a simple predictor with the benefits that it has a low initial computational cost and it can also be easily updated online while it is in use. Any newly observed transition from cell s_i requires only that the transition probabilities from s_i be updated, the transition probabilities of all the other cells need not be modified.

B. A History-based Predictor Using the Expectation-Maximisation Algorithm (HEM)

We note that using a single DTMC, as in the previous method, to represent the mobility pattern of all users is quite unrealistic, since different users will follow very diverse and potentially conflicting movement patterns. In this new *History-based Expectation Maximisation* (HEM) approach, we instead use multiple DTMCs which each represent different classes of user mobility pattern. When we wish to make a prediction, we use a user's recent history to determine which class/DTMC they are currently following. Note that a user is not restricted to permanently following a single mobility class, they may vary their class at any time. For example, in the morning a user may be following a class which represents "*morning rush hour traffic*" when on their way to work. When returning home they may be following a different class of behaviour, "*evening rush hour traffic*".

We define a total of K different DTMCs/classes of user mobility. We use a latent variable model for co-occurrence data, which associates an unobserved class variable $z_k \in Z = \{z_1, z_2, ..., z_K\}$ with each observed occurrence of a cell transition in the training data set, representing which class of mobility a user was in when making that transition. The probability that a user u_a is following mobility class z_k at any given time, is defined as $P(z_k|u_a), \forall z_k \in Z, \forall u_a \in U$.

Each of the different DTMCs have their own respective transition probabilities $P(s_i \rightarrow s_j|z_k), \forall s_i, s_j \in S, \forall z_k \in Z$ representing the probability that a user will transition from cell s_i to cell s_j given they are in mobility class z_k. However, populating these transition probabilities is not as simple as for our previous method. The training data set contains only sequences of cell transitions, providing us with observation pairs $(u_a, s_i \rightarrow s_j)$, but it does not inform us to which mobility class those transitions belonged and we do not know the value of $P(z_k|u_a)$ for any users. Similarly, to calculate values for $P(z_k|u_a)$ from the training data set, we would first need to know the values of $P(s_i \rightarrow s_j|z_k)$.

Inferring values for $P(s_i \rightarrow s_j|z_k)$ and $P(z_k|u_a)$ cannot be done analytically, but can be approximated using the Expectation-Maximisation (EM) algorithm [14]. The EM algorithm is typically used to compute Maximum Likelihood Estimates (MLE) in models with incomplete data or hidden/latent variables, like our current problem. The EM algorithm alternates two steps: (1) An expectation (E) step where posterior probabilities are calculated for the latent variables, based on the current estimates of the parameters,

and (2) a Maximisation (M) step where parameters are updated to maximise the expected complete data log-likelihood, which depends on the posterior probabilities computed in the E step.

Similar to the approach taken by [15], this results in the following Expectation step for our model:

$$P(z_k|u_a, s_i \rightarrow s_j) = \frac{P(s_i \rightarrow s_j|z_k)P(z_k|u_a)}{\sum_{z' \in Z} P(s_i \rightarrow s_j|z')P(z'|u_a)} \quad (1)$$

for all $z_k \in Z, u_a \in U$, and $s_i, s_j \in S$. In the Maximisation step, we then need to maximise the expected complete data log-likelihood, $\mathbb{E}[\mathcal{L}^c]$, which is given by:

$$\mathbb{E}[\mathcal{L}^c] = \sum_{u_a \in U} \sum_{s_x, s_y \in S} n(u_a, s_x \rightarrow s_y) * \sum_{z_k \in Z} P(z_k|u_a, s_x \rightarrow s_y) log[P(s_x \rightarrow s_z|z_k)P(z_k|u_a)] \quad (2)$$

where $n(u_a, s_x \rightarrow s_y)$ is the number of times user u_a made a transition from cell s_x to cell s_y. Maximising $\mathbb{E}[\mathcal{L}^c]$, as in [15], then gives us the two following re-estimation equations to use in the M-step:

$$P(s_i \rightarrow s_j|z_k) = \frac{\sum_{u_a \in U} n(u_a, s_i \rightarrow s_j)P(z_k|u_a, s_i \rightarrow s_j)}{\sum_{s_x, s_y \in S} \sum_{u_a \in U} n(u_a, s_x \rightarrow s_y)P(z_k|u_a, s_x \rightarrow s_y)} \quad (3)$$

$$P(z_k|u_a) = \frac{\sum_{s_x, s_y \in S} n(u_a, s_x \rightarrow s_y)P(z_k|u_a, s_x \rightarrow s_y)}{n(u_a)} \quad (4)$$

where $n(u_a)$ is the total number of cell transitions made by user u_a. To then use the EM algorithm, we initially assign valid randomised values to $P(s_i \rightarrow s_j|z_k)$ and $P(z_k|u_a)$ and then we repeatedly alternate between the E-step [Eq. 1] and the M-step [Eqs. 3+ 4] until convergence to a local maximum. Since the model possesses multiple maxima, we repeatedly re-run the EM algorithm with different initial random values, and store the result with the maximum value for $\mathbb{E}[\mathcal{L}^c]$.

This then gives us a set of K DTMCs defined by $P(s_i \rightarrow s_j|z_k)$. To use the DTMCs to make a prediction for the probability of a user moving to future cell s_f given a history of cell transitions, $H = \{s_1 \rightarrow s_2, s_2 \rightarrow s_3, ..., s_{(c-1)} \rightarrow s_c\}$ where s_c is the current cell of the user, we first give a weighting $W(z_k|H)$ to the different DTMCs, proportional to how likely it is this user was following that mobility class.

$$W(z_k|H) = \frac{\sum_{(s_i \rightarrow s_j) \in H} P(s_i \rightarrow s_j|z_k)}{\sum_{z' \in Z} \sum_{(s_i \rightarrow s_j) \in H} P(s_i \rightarrow s_j|z')} \quad (5)$$

Finally, we aggregate the weighted predictions of each DTMC to predict the probability of transitioning to cell s_f:

$$P(s_c \rightarrow s_f|H) = \sum_{z_k \in Z} W(z_k|H)P(s_c \rightarrow s_f|z_k) \quad (6)$$

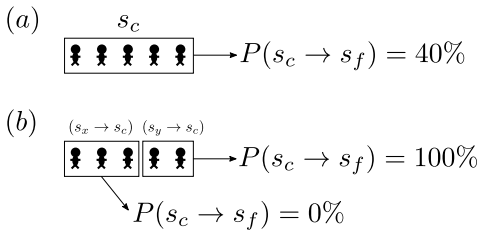

Fig. 2. Two ways of grouping the same 5 people into correlative sets.

This approach scales for any length of history of cells ≥ 1 without the need to recompute the DTMCs. The computational complexity and need to repeatedly re-run the EM algorithm to find the global maximum renders HEM less suitable for online updating than sDTMC, as any changes to the training data set requires that the entire algorithm be re-run in full. Once we have generated the DTMCs, the time to compute a prediction scales linearly with the size of K and the length of the history H but it is only a matter of milliseconds even for very large K and H. Our expected use case for this predictor is to accumulate cell transition data while providing predictions without updating the DTMCs, and then to generate new DTMCs off-line during periods of low user activity.

III. Evaluation of Predictions

There is an important distinction that must be made between how one can evaluate traditional predictions and PDF-predictions. In the traditional view, when presented with a single instance of a user being at location l_c and the predictor predicting a future location l_p, we can determine if the prediction was correct or not by evaluating if the user's actual future location l_f is equal to l_p. However, if we give a PDF-prediction that it is 70% likely the user transitions $(l_c \rightarrow l_p)$, we cannot determine if the prediction was correct or not, regardless of what value l_f takes. To evaluate it correctly, we must first look at the set of features which the predictor takes as input parameters. We require a large set of similar instances which match with respect to the predictor's input features, over which we can then average the users' behaviour and can evaluate whether or not users in that situation will move to l_p 70% of the time or not. We shall refer to such a set of instances as a *Correlative Set*.

For the sDTMC predictor, its input feature is the current cell. For our HEM predictor, its input features are the current cell and the past cells of a user. We note that any correlative set of instances for HEM must also be a correlative set for sDTMC because they must share the same current cell.

The more that the input features restrict the possible members of correlative sets, the more valuable the predictions one can make. Figure 2 illustrates this concept, showing two possible ways of dividing up the same set of people into correlative sets. In Fig. 2(a), the 5 people are grouped into a single correlative set based upon their common feature, the current cell s_c, and there is a 40% probability of users progressing to cell s_f. In Fig. 2(b), the same 5 people are subdivided into two separate correlative sets, based upon input

features of both their current cell s_c and their previous cells, s_x and s_y respectively. The observed values for $P(s_c \rightarrow s_f)$ in (b) are different to the single correlative set of (a). Accurately predicting the observed values of the more restricted correlative sets in (b) would provide more useful predictions than (a), since they are averaging over a less diverse variety of people. However, the more restricted the correlative sets of a predictor, the more challenging it is to fairly evaluate, as data sets will contain fewer instances matching the input features.

We assume that services that use PDF-predictions would treat predictions with close values similarly, and be more focused on distinguishing between more distantly separated values. E.g. 2% and 3% are likely to be handled almost identically since both are very unlikely to occur, while 50% and 75% are far enough apart to need to be treated very differently. Therefore, in our analysis we focus on the absolute difference between our predictions and actual values, rather than the difference as a proportion of the actual value, since as long as our predicted value is within a reasonable range of the actual value, the prediction can be used by services. We expect the tolerable range to vary based on specific services' needs, and we include our performance at the range of 10% in our evaluations as an illustrative example.

IV. Experimental Evaluation

We first evaluate our new algorithm HEM against sDTMC on the MDC [11], [12] data set which contains both pedestrian and vehicular GPS data traces from the Lake Geneva region of Switzerland, gathered over 18 months. Participants' GPS records were gathered in intervals of 10 seconds. To present an illustrative example, we use a restricted region of approx. 1744m x 1909m, corresponding to the approximate area of detection for the base station with id 53577. 146 users provided GPS records in this cell's region. We divided the region into 20x20 small cells, of approx 87m x 95m each. For HEM, we use $K = 10$, as it was found to provide a good balance between off-line computation time and prediction quality.

The data set was preprocessed to improve the consistency of the records. User GPS records were divided into *journeys*: consecutive sequences of GPS records not more than a threshold of 50 seconds apart. If two GPS records within a journey were more than 10 seconds apart, additional intermediate records were interpolated to ensure no more than 10 seconds occurred between any records of a journey. To simplify the modelling process, only cell transitions between adjacent cells were allowed. Consecutive records which transitioned to non-adjacent cells had extra intermediate cells interpolated to ensure only transitions to adjacent cells were possible. After preprocessing, 20% of all journeys were used as training data and the remaining 80% were used as test data for evaluation.

As described in Section III, to fairly evaluate a prediction a large number of similar instances is required, so we filter out all correlative sets containing less than 100 instances in the test data. Figure 3 shows the average absolute difference (AD) and and the mean squared error (MSE) between the predicted and observed probabilities for each distinct correlative set, for the two approaches. The X axis shows the length of the history, $|H|$, which define the correlative sets used for evaluation: at each plot we evaluate for all sets of instances with every possible history of the given length.

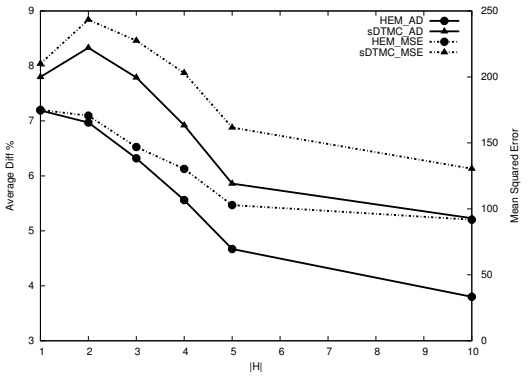

Fig. 3. The average difference between predictions and the MDC data set.

sDTMC makes decisions independent of history, but altering $|H|$ changes the correlative sets we are comparing it against, and so its performance is not constant. Since HEM takes account of history, it outperforms sDTMC at all values of $|H|$. At $|H| = 1$ where we perform the worst, HEM is on average 7.19% away from the observed value, which for most services is more than sufficiently accurate. Our accuracy steadily increases as $|H|$ rises, meaning that our predictions are simultaneously becoming more useful and more accurate. This can partly be attributed to the increasing proportion of histories in which the user does not change cell for many consecutive time intervals which are easier to predict.

HEM (and sDTMC) provided 9251 predictions for the distinct history patterns which passed the 100 instance filtering threshold. 82% of all predictions by HEM were accurate to within 10% of the observed values. Table I shows a breakdown of the average difference between HEM's predictions and the observed value, grouping predictions into ranges based on their predicted value. HEM's predictions made in the lowest ranges ($\leq 20\%$) and the highest ranges ($\geq 70\%$) have a high accuracy, which may be the easiest predictions for services to use since they are closest to predicting a fixed outcome. However in the $(20\% - 70\%)$ range our predictions' reliability drops. At $|H| = 1$, the average difference in the $(90\% - 100\%)$ range is abnormally high because only 1 correlative set was predicted in this range. At $|H| = 10$, HEM did not make any predictions in the $(20\% - 60\%)$ range. However, there were only 2 correlative sets with observed values in that range, both located in the $(20\% - 30\%)$ range.

Next, we evaluate the predictors against Crawdad's [13] data traces of 320 taxis in Rome, Italy, collected over 1 month. The traces contain GPS records accurate to within 20m, recorded every 7 seconds. Due to the large size, we focused on the most densely populated 500m x 500m region, containing 951952 GPS records. We again divided this region into 20x20 cells of 25m x 25m dimensions, and used 20% of journeys for training.

Figure 4 shows the AD and MSE on the Crawdad Rome

TABLE I
BREAKDOWN OF HEM'S PREDICTIONS' AVERAGE DIFFERENCE ON THE MDC DATA SET, GROUPED BY PREDICTION VALUE.

$\|H\|$	0-10%	10-20%	20-30%	30-40%	40-50%
1	4.15%	8.27%	17.74%	20.26%	20.19%
2	4.14%	8.18%	17.02%	20.52%	21.04%
3	3.49%	7.78%	19.68%	20.33%	21.61%
4	3.49%	6.88%	19.95%	23.84%	15.92%
5	3.02%	6.44%	19.34%	34.15%	19.01%
10	2.88%	5.83%	-	-	-
$\|H\|$	50-60%	60-70%	70-80%	80-90%	90-100%
1	16.42%	12.26%	11.33%	11.71%	91.37%
2	17.64%	10.37%	9.44%	11.18%	9.81%
3	16.62%	10.31%	8.36%	6.80%	10.18%
4	14.56%	8.64%	8.06%	2.67%	10.92%
5	10.96%	10.07%	6.98%	2.99%	9.26%
10	-	15.97%	6.85%	2.73%	8.29%

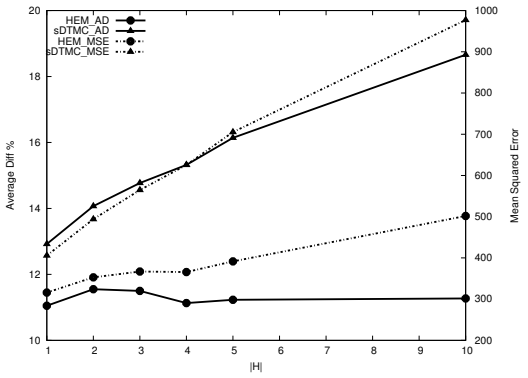

Fig. 4. The average difference between predictions and the Crawdad data set.

taxi data set. The performance of sDTMC decreases as we increase the size of $|H|$. Because their destinations are dictated by passengers, the taxis move more unpredictably than the MDC participants who had more fixed routines. sDTMC cannot be used to provide reliable predictions for the vehicular movement exhibited in this data set. HEM also performs worse than in the MDC data set, but far more stably than sDTMC, with AD remaining almost constant at 11%, and MSE rising more slowly. On this denser Crawdad data set, HEM provided 39871 predictions and was accurate to within 10% for 65% of them. Table II shows the breakdown of HEM's successful predictions. The overall trend is the same as the MDC data: very good accuracy at the lowest and highest ranges, but much poorer performance in the middle ranges of $(20\% - 80\%)$.

A. Improving predictions with a more restricted correlative set

For our final experiments, we investigated the use of a more restricted correlative set to provide more useful predictions, based upon information at the GPS coordinate level rather than at the cell level. Using GPS records, the user's future cell is predicted as being the cell they would reach if they maintain the same velocity as between their previous and current coordinates. We identified a scenario, $scen_\alpha$, in this type of prediction proved highly accurate: when a user's future GPS-predicted, current and previous cells were identical, 92% of the time the user was observed to remain in the same cell.

TABLE II
BREAKDOWN OF HEM'S PREDICTIONS' AVERAGE DIFFERENCE ON THE CRAWDAD DATA SET, GROUPED BY PREDICTION VALUE.

$\|H\|$	0-10%	10-20%	20-30%	30-40%	40-50%
1	4.93%	14.71%	20.57%	25.99%	25.68%
2	5.32%	15.08%	20.71%	26.85%	26.98%
3	4.71%	14.32%	21.94%	28.86%	29.79%
4	4.12%	13.83%	21.82%	30.07%	29.92%
5	3.68%	14.02%	24.33%	31.80%	31.82%
10	2.05%	20.10%	35.39%	45.32%	45.01%
$\|H\|$	50-60%	60-70%	70-80%	80-90%	90-100%
1	22.16%	22.16%	14.20%	10.09%	6.81%
2	23.72%	24.96%	16.39%	13.17%	11.96%
3	22.67%	22.40%	17.30%	8.96%	7.11%
4	24.24%	22.80%	16.59%	8.13%	7.80%
5	26.00%	19.96%	16.09%	6.61%	6.25%
10	38.32%	28.42%	20.94%	7.23%	3.45%

We then developed a new PDF-predictor, GPS-Stationary (G-Stat), which functions as a hybrid with either HEM or sDTMC as follows: for each cell s_i, we calculate the accuracy of the GPS-based prediction, acc_i, for all occurrences of $scen_\alpha$ in the training set. If a cell s_i has no instances in the training data, we set $acc_i = 0.92$, which was the average accuracy of all cells which had instances in the training data. Then we define: $P(s_i{-}{>}\, s_i|scen_\alpha) = acc_i, \forall s_i \in S$.

For sDTMC:

$$P(s_i{-}{>}\, s_j|scen_\alpha) = \frac{(1-acc_i)P(s_i{-}{>}\, s_j)}{\sum\limits_{\forall s' \in S, s' \neq s_i} P(s_i{-}{>}\, s')}$$

For HEM:

$$P(s_i{-}{>}\, s_j|scen_\alpha) = \frac{(1-acc_i)P(s_i{-}{>}\, s_j|H = (s_i{-}{>}\, s_i))}{\sum\limits_{\forall s' \in S, s' \neq s_i} P(s_i{-}{>}\, s'|H = (s_i{-}{>}\, s_i)}$$

for all $s_j \in S, s_j \neq s_i$. All instances not satisfying $scen_\alpha$ are handled with versions of sDTMC/HEM trained on data with all occurrences of $scen_{alpha}$ filtered out. Figure 5 shows the AD for both variants of G-Stat on both the MDC and Crawdad data sets. HEM's MSE was consistently the best but is omitted for clarity. G-Stat(sDTMC) shows the largest improvement over its un-enhanced version, but never surpasses G-Stat(HEM). G-Stat(HEM) provided 10856 predictions for MDC and 40860 for Crawdad, with 83% and 70% of predictions being within a 10% range of the observed values, respectively. G-Stat(HEM) shows a small but consistent improvement over HEM in all data plots, performing best on MDC at $|H| = 10$ where it has only 2.2% average inaccuracy. In addition to increased accuracy, G-Stat provides more useful predictions by operating on more restricted correlative sets.

V. RELATED WORK

Cheng et al. [6] provide a review of order-k Markov predictors and text compression-based predictors. The order-k Markov predictors described are restrictive in that they require a history of exactly length k to function. Markov chains continue to be a popular method of location prediction, recently [1] used their predictions to reduce handovers in future cellular systems that possess a logical separation between data and

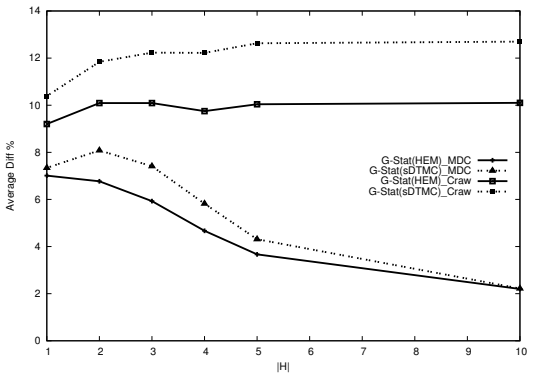

Fig. 5. The avg diff for both variants of G-Stat on the MDC and Crawdad data sets.

control planes. The authors of [16] use 2nd-order Markov chains to predict both the user's destination and their most likely path to that destination. Some service focused papers, like [4] which uses Hidden Markov Models to assign users to Femtocell Access Points (FAP), appear to implicitly be making use of PDF-predictions to influence their choices. However because they are focused on measuring the effect of predictions on the performance of a specific service, they do not provide a direct analysis of the accuracy of their PDF-predictions. This does reinforce that there is a tangible need for PDF-predictors for use by services. Andrei et al. [17] used an Expectation-Maximisation [14] (EM) algorithm to generate DTMCs that classify user activity patterns for their iOS app, Hungry Yoshi. The approach taken is very similar to what we use to generate our HEM predictor. However, they used probabilistic temporal logics to analyse and gain insight into users software usage patterns, but did not attempt to predict any users' future actions in the app. [7] also use the EM algorithm for modeling human location data. They apply kernel density estimation using a mixture model approach. They use their models to detect unexpected geolocation events for recognising identity thefts.

VI. Conclusions

We have presented a new probability distribution-format approach to providing mobility predictions, targeting the shortcomings of existing predictions which restrict services' potential functionality. PDF-predictions greatly increase services' control, but at the cost of increased difficulty in evaluating the accuracy of predictors. We have developed a framework for the evaluation of PDF-predictors and have developed two PDF-predictors, HEM and G-Stat, and evaluated their performance when predicting future cells for mobile users.

We have shown that predictions with HEM are very accurate at predicting very low and very high probabilities, which constitute the majority of all observed predictions, but are weaker at predicting values in the middle ranges. By augmenting HEM with G-Stat, we operated on more restricted correlative sets and provided a greater degree of accuracy as well as more useful types of predictions.

In future work, we wish to develop a wider variety of predictors that make use of a more diverse set of input features, such as time, gender, age, speed, etc. and to apply them to specific services, to show the benefits of using PDF-predictions over traditional predictions to influence decisions.

VII. Acknowledgements

Portions of the research in this paper used the MDC Database made available by Idiap Research Institute, Switzerland and owned by Nokia. This work is supported by grant number 10/CE/I 1853 from Science Foundation Ireland, as part of CTVR.

References

[1] A. Mohamed, O. Onireti, S.A. Hoseinitabatabaei, M. Imran, A. Imran, R. Tafazolli, *Mobility prediction for handover management in cellular networks with control/data separation*, in Communications (ICC), 2015 IEEE International Conference on, pp.3939-3944, June 2015 doi: 10.1109/ICC.2015.7248939

[2] V.A. Siris, M. Anagnostopoulou, *Performance and energy efficiency of mobile data offloading with mobility prediction and prefetching.*, in World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a, pp.1-6, June 2013 doi: 10.1109/WoWMoM.2013.6583450

[3] D. Stynes, K. N. Brown, C. Sreenan, *Using opportunistic caching to improve the efficiency of handover in LTE with a PON access network backhaul*, 2014 IEEE 20th International Workshop on Local & Metropolitan Area Networks (LANMAN), pp.1-6, May 2014 doi: 10.1109/LANMAN.2014.7028621

[4] A. B. Cheikh, M. Ayari, R. Langar, G. Pujolle, L. A. Saidane, *Optimized Handoff with Mobility Prediction Scheme Using HMM for Femtocell Networks*, in Communications (ICC), 2015 IEEE International Conference on, pp.3448-3453, June 2015 doi: 10.1109/ICC.2015.7248858

[5] A. Noulas, S. Scellato, N. Lathia, C. Mascolo, *Mining User Mobility Features for Next Place Prediction in Location-Based Services*, Proceedings of the 2012 IEEE 12th International Conference on Data Mining (ICDM'12), pp.1038-1043, 2012 doi: 10.1109/ICDM.2012.113

[6] C. Cheng, R. Jain, E. v. d. Berg, *Location Prediction Algorithms for Mobile Wireless Systems*, Wireless Internet Handbook, M. Illyas and B. Furht, Eds. CRC Press, pp.245-263, 2003

[7] M. Lichman, P. Smyth, *Modeling Human Location Data with Mixtures of Kernel Densities*, Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'14), pp.35-44, 2014 doi: 10.1145/2623330.2623681

[8] H. Jeung, H.T. Shen, X. Zhou *Mining Trajectory Patterns Using Hidden Markov Models*, Data Warehousing and Knowledge Discovery, pp.470-480, 2007 doi: 10.1007/978-3-540-74553-2_44

[9] https://www.uber.com/

[10] https://www.lyft.com/

[11] N. Kiukkonen, J. Blom, O. Dousse, D. Gatica-Perez, J. Laurila, *Towards Rich Mobile Phone Datasets: Lausanne Data Collection Campaign*, in Proc. ACM Intn. Conf. on Pervasive Services (ICPS), Berlin, July 2010

[12] J. K. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T. Do, O. Dousse, J. Eberle, M. Miettinen, *The Mobile Data Collection Challenge: Big Data for Mobile Computing Research*, in Proc. Mobile Data Challenge Workshop (MDC) in conjunction with Int. Conf. on Pervasive Computing, Newcastle, June 20113

[13] http://crawdad.org/roma/taxi/20140717/

[14] A. P. Dempster, N. M. Laird, D. B. Rubin, *Maximum Likelihood from Incomplete Data via the EM Algorithm*, Journal of the Royal Statistical Society, Series B, Vol. 31, No. 1, pp.1-38, 1977 doi: 10.2307/2984875

[15] T. Hofmann, *Unsupervised Learning by Probabilistic Latent Semantic Analysis*, Journal of Machine Learning, Vol. 42, No. 1-2, pp.177-196, January-February 2001 doi: 10.1023/A:1007617005950

[16] A. Nadembega, A. Hafid, T. Taleb, *A Destination and Mobility Path Prediction Scheme for Mobile Networks*, in Vehicular Technology, IEEE Transactions on , vol.64, no.6, pp.2577-2590, June 2015 doi: 10.1109/TVT.2014.2345263

[17] O. Andrei, M. Calder, M. Higgs, M. Girolami, *Probabilistic Model Checking of DTMC Models of User Activity Patterns*, 11th International Conference on Quantitative Evaluation of Systems (QEST2014), pp.138-153, September 2014 doi: 10.1007/978-3-319-10696-0_11