Where will you go? Mobile Data Mining for Next Place Prediction

João Bártolo Gomes[1], Clifton Phua[2], Shonali Krishnaswamy[1]

[1] Institute for Infocomm Research (I2R), A*STAR, Singapore
{bartologjp, spkrishna}@i2r.a-star.edu.sg
[2] SAS, Singapore clifton.phua@sas.com

Abstract. The technological advances in smartphones and their widespread use has resulted in the big volume and varied types of mobile data which we have today. Location prediction through mobile data mining leverages such big data in applications such as traffic planning, location-based advertising, intelligent resource allocation; as well as in recommender services including the very popular Apple Siri or Google Now. This paper, focuses on the challenging problem of predicting the next location of a mobile user given data on his or her current location. In this work, we propose *NextLocation* - a personalised mobile data mining framework - that not only uses spatial and temporal data but also other contextual data such as `accelerometer`, `bluetooth` and `call/sms` log. In addition, the proposed framework represents a new paradigm for privacy-preserving next place prediction as the mobile phone data is not shared without user permission. Experiments have been performed using data from the Nokia Mobile Data Challenge (MDC). The results on MDC data show great variability in predictive accuracy of about 17% across users. For example, irregular users are very difficult to predict while for more regular users it is possible to achieve more than 80% accuracy. When compared against existing results, our approach achieves the highest predictive accuracy. Finally, we propose an alternative business model for mobile advertising that uses *NextLocation* framework.

1 Introduction

Next place prediction is a particular problem of location prediction where the challenge consists of predicting the next location of a mobile user given his current location [15, 11]. Most existing work models next place prediction as a classification problem, where spatial and temporal data is used for training.

However, issues such as the integration of other rich contextual data, available on smartphones nowadays such as `accelerometer`, `bluetooth` and `call/sms` logs have not been seriously investigated. In addition, most existing approaches focus mainly on the classification problem assuming the data in a centralised server while other problem specific issues related to user behavioural changes, privacy, data management and scalability have not been explored in-depth.

To address these issues in this paper, we propose *NextLocation* - a novel integrated framework for the next place prediction problem - that predicts the next location using only current location and contextual data for each mobile phone user. *NextLocation* learns an "anytime" classification model which incorporates past data to predict the next place in an incremental manner. It enables greater personalisation and privacy while bringing the whole learning process on-board the mobile device. Moreover, in addition to spatial and temporal information, the proposed approach combines other context information available on the mobile device. The main advantages of the *NextLocation* for next place prediction are:

- Privacy-preserving as this is a key user need to be in control of the their personal mobile phone data [13, 7]. Using *NextLocation* for mobile data mining, personal mobile phone data is not shared with an external party without permission.

– Reduced communication overheads in terms of bandwidth as well as battery drain since local processing is usually less expensive than wireless data transfer [18].
– Dynamic instead of static model building facilitates the model adaptation so it reflects up-to-date user behaviour.
– Allows online estimation of personalised next place predictive accuracy.
– Enables an alternative business model where advertisement providers can push content that is relevant to a certain location and the user will only receive it when is about to visit it.

The rest of the paper is organised as follows. The following Section reviews the related work. Section 3 presents next place prediction as a classification problem, which is followed by a detailed description of the feature engineering from Nokia Mobile Data Challenge (MDC) in Section 4. The proposed approach for next place prediction (*NextLocation*) is presented Section 5. The experimental setup and results are discussed in Section 6. In Section 7, we propose an alternative business model for mobile advertising that uses the proposed approach. Finally, in Section 8, conclusions of this work and future work are presented.

2 Related Work

Location prediction assumes that mobile sensor observations from wireless local network (Wi-Fi), Global System for Mobile Communications (GSM), Global Positioning System (GPS) are available. The prediction task consists of using such data to know and understand the user's current location. The research on mobile user visiting behaviour, can bring additional value to different domains, such as mobile advertising [1], resource allocation [22] and disaster relief [23].

[19] proposed a general model for semantic trajectories, and introduced the concept of stops and moves. The locations of interest are the locations where the user stops for a period of time and the semantic trajectory represents the visiting history of semantic places (for example, work, home, restaurants). [16] proposed a clustering-based approach to discover the interesting semantic places in trajectories.

In this work, we are interested in a related but more challenging location prediction problem, which aims to predict the next location without knowing in advance the readings from future sensor data. In general, the mobile data used for the next location problem consists of the historical information about the `visit` sequences and associated context information (for example, timestamps, `accelerometer`, `bluetooth` and `call/sms` log) from these visits.

There is extensive research on the problem of predicting future locations. Most of such work creates a model based on frequent patterns and association rules from a history of user/collective trajectories as an ordered sequence of locations that are timestamped [15]. Other sequential learning models such as Hidden Markov Models [14], Conditional Random Fields [17], Particle Filter [4] have been also applied to this problem. However, the problem addressed in this paper is different because, for any user, the prediction of the next location assumes only knowledge about the current location (without data about previous locations). This limited/reduced history makes our problem more general as it is not unusual to have gaps in mobile sensor data. *Gaps* refer to significant time periods where the mobile phone is not collecting data (for example, when the mobile phone has run out of battery).

Recently the Nokia Mobile Data Challenge (MDC) released a large dataset for research and one of the dedicated tasks consisted of next place prediction [11]. From this MDC challenge, several approaches were able to predict the next place with high accuracy [3, 21, 8, 12, 20]. The proposed approaches focused on learning a model for each user which captures the spatio-temporal trajectory of user visits. Great effort was dedicated to feature engineering for each approach.

Still, two main issues remain relatively unexplored in the literature of next place prediction. First, privacy issues arise from using such data, although there are efforts in the direction of anonymization [11]. Second, rich context information can be exploited for personalisation. In this paper, we try to address these issues by proposing a mobile data mining framework that does not require the raw data to be disclosed and that the model built is highly personalised.

3 Next Place Prediction: Definition

First let us assume we are interested in finding the next destination of a single user when (s)he is still at the current location. It is easy to generalise from this problem to multiple users. Consider $L = \{l_1, ..., l_n\}$ to be the set of values of visited (for at least a certain amount of time) spatial locations, the $T = \{t_1, ..., t_n\}$ to be the set of timestamps and $C = \{c_1, ..., c_n\}$ to be the set of context information where c_i represents itself a set of attribute value pairs that are in available at t_i. This context information is usually the data available in the user's mobile phone and can be collected from the `accelerometer`, `bluetooth`, `call/sms` log, `wlan` (Wi-Fi) or `phone status` (consider that for some users charging the phone is only performed at certain locations).

Given a series of historical visits to different locations in the past, that constitutes the data available for training $H = \{(L, C, T)\} = \{(l_1, c_1, t_1), ..., (l_k, c_j, t_j)\}$ and the context $C = ctx(t_i)$ at $T = t_i$ of the latest location $L = loc(t_i)$, the next place prediction problem can be formulated as finding the most likely location

$$argmax_{l \in L} \left(p(L_{next} = l | T = t_i, C = ctx(t_i), L = loc(t_i)) \right)$$

Please note that the prior only considers the current location and not the past location or a sequence of previous locations as is usually modelled using Hidden Markov Models (HMM) [14] or Conditional Random Fields (CRF) [17]. The reason for this is simple, it is not always possible to have a sequence of visits without gaps, therefore, we prefer to define the more general problem where we are able to make a prediction if we at least know the current location.

In Section 6 we will describe in detail a particular instantiation of the problem, the feature engineering process, and report and discuss the results of our experiments with real data.

4 Mobile Data Challenges

In this section, we discuss the challenges that come from collecting mobile data for the next place prediction problem. Understanding the whole data process and its requirements allowed us to design and explore the alternative solution proposed in this paper. The following subsections describe what we consider some of the challenges that need to be addressed to transform the data available in the mobile phone into a format that can be used to induce a model useful for next place prediction as defined previously.

Location detection The raw data of each user's location is usually estimated based on GPS and Wi-Fi that is then transformed into a semantic place which captures most of the mobility/location-based information without including the actual geographic coordinates/access points. Moreover, information from social networking services that support location, such as Four-Square, Facebook or Google Latitude already allows the user to 'check in'. These services already include automatic location detection which can be leveraged to create or enrich the temporal series of semantic locations, that we require for next place prediction. Therefore,

this paper will not focus on semantic place (for example, place tagged as home, workplace, or transportation place) prediction but on next place prediction that we formally define in Section 3.

User Specificity Next place prediction is a user specific problem as the set of locations visited is personal and even if this set might overlap among different users the trajectory of user visits to different location is most likely unique. It is therefore, hard or impossible to accurately learn joint models over multiple users as can be performed in other classification tasks such as activity/speech recognition. The user specificity challenges motivates the usage of a personalised model.

Evolving data The user movement behaviour might change over time. For instance, changing house/city/country/workplace can have a profound impact on the most recent movement pattern. Therefore, we propose that modelling should be adaptive and the usage of an incremental anytime model, that incorporates new information and forgets old outdated information. Moreover, the model should incorporate novel locations seamlessly.

Sparse and missing data It is possible to have missing data or gaps in the sequence of visits to particular locations. This is the main reason that led us to formulate the next place prediction problem considering only the current location and not a sequence of past locations that precede the current location in time. This challenge is related with model evaluation as the number of observations (evidence that from location l_i the user moved to location l_j) and how representative they are of user mobility patterns will have a high impact on the accuracy of the learnt model.

5 Next Place Prediction: *NextLocation*

The framework proposed in this paper, that we call *NextLocation*, models next place prediction as a classification problem. However, instead of executing the traditional learning process (i.e., data collection, data transfer, model building, model deployment), we create an integrated framework that is executed on the mobile device itself.

One key innovation of *NextLocation* is that it preserves user privacy as it allows the building of a model for next place prediction without disclosure of private user data. Such framework gives the user control over who can use this model results (i.e., the next location predictions) without disclosing the real locations visited and associated context data.

Figure 1 illustrates *NextLocation* learning process and its components. We can observe that the pre-processing component, anytime model, and accuracy estimator play a central role in the proposed framework. Each of these components perform the following:

- Pre-Processing - the raw data must be pre-processed/transformed for next place prediction. Here, the location data from a visit is enriched with other context information. The pre-processing component only requires to keep a short term window of data. When updating the model, the data represents the previous visit location and its context information, and the target variable (to predict) is the current location.
- Anytime Model - must be able to integrate new information as it is available (such as new visits) and must be also able to predict the next location. Any classification algorithm that to learns incrementally can in principle be used in this component to create/update the anytime model. Moreover, these algorithms are light-weight and can be executed using the computational resources usually available on current smartphones. However, it is beneficial if the algorithm can adapt the anytime model when there is evolution in the observed data.
- Accuracy Estimator - comparing the anytime model prediction with the actual destination allows to keep an estimate of next place prediction accuracy.

Past data can be discarded once it is incorporated into the anytime model, consequently, the memory consumption of the *NextLocation* learning process is very low compared to approaches that require to collect all data and process it in batch mode.

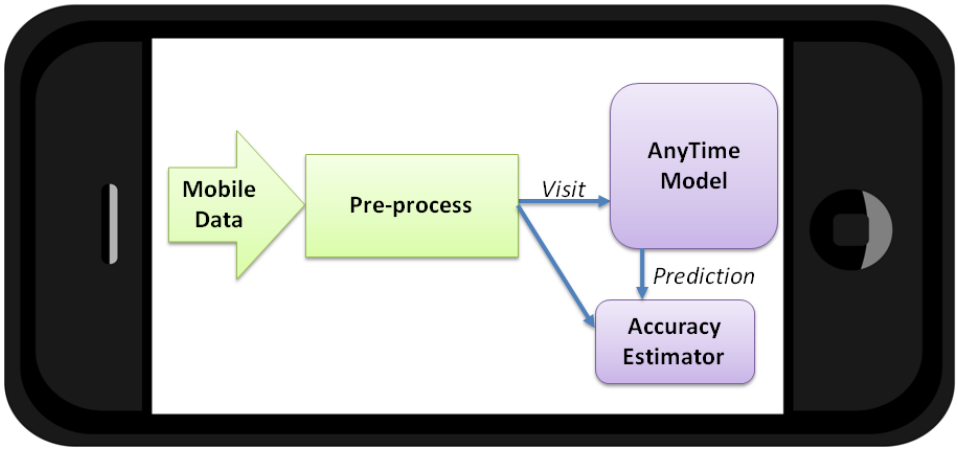

Fig. 1: *NextLocation*: framework overview

Adapting the model Given the issue of data evolution it is important to adapt the anytime model. For instance, adapting the model to a new living environments that causes a change in the user mobility patterns. In such situations it is likely that the most recent past represents the activities of interest and less importance should be given to older records that represent past behaviour. The simplest solution to achieve model adaptation is that the anytime model represents only the most recent records that belong to a sliding window of fixed size or weigh the records accordingly to their age. In our experiments we preliminarely explored more sophisticated approaches such as drift detection (detect and adapt to changes in the data) but we plan to study the adaptation issue in more detail in further research.

On-line Model Evaluation As part of the proposed framework we keep an estimate of the anytime model accuracy online. Here we briefly formalise the evaluation procedure. The prequential-error [6] is computed based on an accumulated sum of a loss function L between the anytime model prediction $\widehat{l_i}$ and the location that is visited next l_i. Note that the prequential error estimated over the entire learning process can be strongly influenced by the first part of the error sequence, where only a small number of records has been processed by the learning algorithm. Therefore, the estimate of the model accuracy can also be represented over a sliding window or using fading factors instead of the whole learning process.

6 Experimental evaluation

This section describes the experiments that were performed to evaluate *NextLocation* approach feasibility and accuracy. The data used in the experiments has been released for the Nokia Mobile Data Challenge (MDC) [11], and was collected from the smartphones of almost 200 participants over the course of over one year in a real world environment.

There was a significant effort into data transformation/feature engineering. We used a total of 70 features including: 11 temporal features, 8 `accelerometer` features, 2 `bluetooth` features, 23 `calllog` (call/sms) features, 20 `visit` related features, 6 `system` features.

6.1 Nokia MDC Dataset

The MDC data was collected on a 24/7 basis over months. In the Dedicated Track of this competition, which included the task of next place prediction, the raw location data is transformed into the sequence of visits to symbolic places.

The users in MDC data are sampled into three separate sets. The training data set (called setA) consists of mobile phone data collected from 80 persons during a period of time varying from a few weeks to two years. The unseen data for each participant in setA is used to build the test data set (called setC), where the unseen data corresponds to the continuation of setA (in time). The setC ground truth was never released after the challenge. However, the a validation set (called setB)was released and is used to evaluate the results. The validation set contains visits that were were randomly chosen from the last part of setA (in time). The validation set was built by filtering data in setA with time intervals corresponding to the randomly chosen visits.

For the MDC challenge, participants were free to estimate the context from all the available data within a determined time interval (i.e., current location corresponding to a visit in a place). The visits were timestamped with the start/end point entering/leaving the location visited. *Trusted* visits (provided with raw features in the form of trusted_start, trusted_end in the `visit_sequence` tables) are more reliable than untrusted visits. For this task only visits where the mobile user stays in that location for 20 or more minutes are considered. Moreover, information about if the transition for that visit location is to be trusted or not is available (i.e., reliable sensor data).

The MDC database has 5 main types of data: environmental, personal, phone usage, phone status, and visits data. This data is represented across 18 tables, with more than 130 raw attributes, and is approximately 50 GB in size. Our focus for this paper is on Task 2, using only the labels for next place prediction. A detailed description of the data collection campaign is available in [11].

A significant challenge that we observed when working with this data, was the fact that while some users had highly regular patterns of movement, for some users there was significant variability. Clearly, the former mobile users have a higher predictability of movement, than the latter. This is further compounded by the fact that some mobile users have significantly more data than the others (though it must be said that more data does not necessarily in this case imply higher predictability).

6.2 Data transformation

In the MDC dataset the transformation of raw location data into a sequence visits (each visit is more than 20 minutes as provided in the challenge) to symbolic places was already processed. The sequence of visits is timestamped is the key data for next place prediction and is similar to what has been proposed in [19]. However, other context information that might be used improve the predictive performance of needs to be derived from the raw data associated to those visits. In the our experiments we ended up with 70 features. In this section we describe our feature extraction process. We would like to note that these features were all calculated per user and using a sliding window approach, this is, the raw data is processed locally and then discarded without the need to keep all the information in main

memory. A frequency table with statistics about the different locations is also kept to calculate more sophisticated features (such as from `bluetooth`).

Temporal features From the start and end timestamps of a particular visit several temporal features were generated. The duration of the visit, the day of the week, weekend or workday, period of the day in two different sets: (AM/PM) ; (morning, afternoon, evening, night). hour of the day (0h-24h). These features can be calculated from both the start and end timestamp.

Phone status features Several types of data about the phone status and the phone operating system was recorded. From this data we derived features to capture the phone status that was characteristic of the visit to a particular location. The most frequent profile (general, silent), the most frequent ring tone used (normal, silent), minimum and maximum battery level, phone charging status, maximum inactive time.

Phone usage features From the phone usage, we consider the information available in the call log, in particular, the most frequent number. We expect that this might help us to capture situations where our next destination is highly correlated with receiving a certain call or text. Usually, before a mobile user leaves the current location for the next destination, the last call or SMS is quite predictive of the next destination (for example, the mobile user calls the person who(s) he will meet later in the next destination). The features generated where the most frequent number: overall, in a call, in a text, in an incoming/outgoing overall, in an incoming/outgoing call, in an incoming/outgoing text, missed call, and the same features calculated but instead of the most frequent the last observation (e.g., last number called, last text sent). From the last call we calculate its duration and if it is an incoming or outgoing call. In addition, we calculate the number of: missed calls, incoming/outgoing calls, incoming/outgoing texts.

Environmental features For the environmental features we explored data from 4 different sensors, `accelerometer`, `bluetooth`, `wlan` and `gsm`. However, since the data is anonymized per user it was impossible to capture information across users. For instance, if two users are in contact with the same GSM tower or Wi-Fi access point the hashed values or the corresponding cell tower ID and access point mac address will appear different despite being the same physical object. Therefore, we used information that is personalised and for which the hash key correspond to the same object that might capture some useful information to the mobile user destination. As environmental features we used:

- `bluetooth`: Similarly to the motivation behind the features we have generated from the call log we tried to understand if there for a certain bluetooth device nearby that influences the next place. We generated one feature that requires some statistics about the current location and observed bluetooth mac addresses for the location. The process tries to calculate the likelihood a certain mac address in the current location is associated with a particular destination.
- `accelerometer`: Accelerometer features that might help to characterise the activity at a given location [9]. This captures a different type of activity compared to the phone status inactivity feature. For instance, there might be situations with no interaction with the phone but since the phone is being carried by the mobile user the accelerometer registers movement. In other situations, the accelerometer registers no movement at all. The features used are: the minimum, maximum, average and standard deviation of the 3 axis accelerometer vector norm captured during the whole visit period and during the last 10 minutes.

Frequency tables Finally for situations where the amount of data of a given user is scarce, we keep a simple frequency table of the most frequent destination and the most frequent destination given the possible temporal features.

6.3 Techniques

In this work, we evaluated different classification techniques to compare across the different models built. In our research, we have employed WEKA [10], a popular suite of data mining software, to benchmark different classification techniques. We have also explored Massive Online Analysis (MOA) [2] - an open-source framework for data stream mining written in Java. Related to the WEKA project [10], it includes a collection of machine learning algorithms and evaluation tools (e.g., prequential-error [6]) particular to data stream learning problems.

Using Weka We performed evaluation of the predictive accuracy using the validation set on the training data. We decided to explore several classification algorithms and our preliminary results indicated a slightly superior performance of the J48 algorithm for decision tree induction. However, our understanding while working with on this problem is that the quality of the instances (i.e., observations) and features that describe them are the most important factors to achieve high predictive accuracy. Consequently, we studied feature selection and instance weighting.

- Feature Selection - As final step after feature engineering, we performed feature selection. This involved using well-known techniques for identifying/ranking which features have the best ability to predict the next location based on the subject's current location.
  We select dynamically for each user the best features (out of the 70 that we constructed/used) using two well-known feature selection techniques from the WEKA. First, information gain and second, cross-validated best feature subset evaluation (CfsSubsetEval). Therefore, the set of features that is selected for each user is different according to their productiveness for that given user/context.
- Instance Weighting - Another issue that we are faced in next place prediction is the quality of the observations, this is, the uncertainty associated with them (due to sensor reading uncertainty/unavailability) and also how relevantly they represent the user mobility patterns. Since the data has information about the uncertainty (a flag associated with a trusted visit, start and end time), we decided to explore this information and perform instance weighting in function of the confidence for their trusted start and end time.

Using MOA In MOA [2] we preliminarily explored different algorithms and obtained good results with Hoeffding Trees. Because of the data evolution issue described in this paper, we decided to explore with a drift detection technique (SingleClassifierDrift). This algorithm implements a well known drift detection method proposed in [5]. Because of the good preliminary results with Hoeffding Trees it was used as the base learner parameter, using other Hoeffding tree classifier variations we obtained similar results.

6.4 Results and discussion

The results presented in this section measure the accuracy on the validation set. This allowed us to compare our approach to existing published results. In Figure 2, *uid* stands for user ID and it does not run in sequence. We can observe that the accuracy for each user on task can have high variance. The results also showed that some more irregular users are very difficult to predict while for some regular users is possible to achieve more than 80% accuracy. We should also note that the some results are biased negatively as these users have a very short history of visits.

Feature Selection In Table 1, we can see the results of our experiments with feature selection. We can observe that in general most features seem relevant to the next place prediction. Consequently, since the feature selection process was

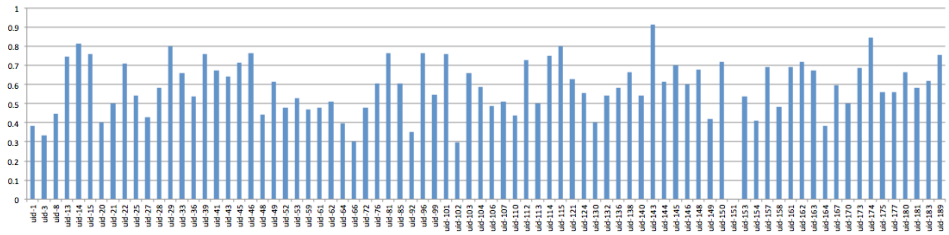

Fig. 2: Predictive accuracy across users

performed per user we make sure that the selection process was personalised. We can see that keeping almost all the 70 features (92%) seems to give the best results. If the set is reduced further a minor decrease in predictive performance is experienced. This finding is interesting as it shows that our effort to build sophisticated features can bring additional predictive accuracy to next place prediction. As feature selection is not so useful for our already predictive features, the results from the subsequent experiments use all the features (without any feature selection).

NFeat%	35%	71%	85%	92%	100%
Accuracy	58.1%	58.8%	59.51%	**59.54%**	59.4%
$Weight_T$	0.0	0.25	0.50	0.75	1.0
Accuracy	57.7%	59.2%	59.3%	59.1%	**59.4%**
$Weight_S$	0.0	0.0	0.5	0.5	0.7
$Weight_E$	0.0	0.5	0.5	0.7	0.5
Accuracy	58.5%	59.4%	59.57%	59.48%	**59.6%**

Table 1: Accuracy with Feature selection and Instance Weighting

Instance Weighting In Table 1, we can see the results of our experiments with instance weighting based on trusted transition. The weight assigned to the instances in case they belong to a trusted transition is determined by $Weight_T$. We can observe that the results are similar among the experiments that still consider the trusted transitions. However, not including weights for trusted transition instances ($Weight_T = 0.0$) will have a significant impact on performance. This can be a consequence of the high number of untrusted transitions (42% of the data or 21356 visits) - in general, more data will be helpful.

In Table 1, different weights are assigned to the transactions based on the trusted start/end flag. Again not including untrusted transitions affects the performance as less instances are available for training. The combination of weights on trusted start time $Weight_S = 0.7$ and weights on trusted end time $Weight_E = 0.5$ gives the best overall results on the validation dataset.

Comparing with Nokia MDC best results Here we compare our best results (OurBest uses all 70 features, has $Weight_T = 1$, $Weight_S = 0.7$, and $Weight_E = 0.5$) with the best results published. Table 2 summarises the best predictive accuracies of 5 other methods (the first three from winning teams). We should note that the ones with asterisk (*) indicate that the reported predictive accuracy was using a different evaluation strategy, and their results have likely overfitted the training data. This happens in results with a Artificial Neural Network (ANN) proposed in [3] (60.83% in validation set with a significantly lower 56.22%

Method	Validation	Competition
ANN [3]	*60.83%**	56.22%
SVM [21]	55.69%	52.83%
HPHD [8]	50.53%	52.42%
Ensemble [12]	55.3%	-
DecisionTree [20]	*61.11%**	-
OurBest	**59.6%**	-

Table 2: Comparision with Nokia MDC results

in competition's test set) and in the J48 DecisionTree approach proposed in [20], where the authors use their own test set as opposed to the proposed validation set for the Nokia MDC.

From the results that can be comparable (without the asterisk) we can see that our best results achieve the highest accuracy. This may be due to the large effort put in feature generation as not a big difference was observed among different techniques.

Online learning Here we report experiments with in MOA that its SingleClassifierDrift algorithm. Evaluating for the same validation set we obtained an average accuracy of 42.22%. Again for some more predictable users it was possible to get close to 80% while for one user was not possible to predict anything. When we compare the results with our best batch results in Figure 2, the batch approach achieves better accuracy overall but that for a small amount of users the results are better with the incremental approach. The batch approach is on average (per user) 17% better than the online approach. Still, when comparing the accuracy with the published results for this dataset, the online approach is still very competitive.

7 Alternative mobile advertisement model

The approach proposed in this paper, *NextLocation*, can be used to support an alternative model advertisement model that we describe in this Section. One of the main ideas in the proposed model is that content is more relevant to a user for a certain location at a certain time. For instance, a user might be interested in dinner promotions before he is about to visit a certain mall at dinner time.

The alternative model consists of 3 parties. The users that are the target of mobile advertisements, the telco provider to which the users subscribe, and the advertisement providers that want to push advertisement content to mobile phone users. Figure 3 illustrates the proposed advertisement model. On the center of the figure we can see the telco provider that receives content $C(l, t, d)$ which in this illustrative model is associated with a location l, time t, and duration d. More meta-data can be used to describe the content (i.e., type) but for simplicity we limit this description to space and time. The users can share with the telco provider $N(l, t, a)$ which is associated with the next location l at time t and estimated accuracy for that prediction a. The telco provider serves as a broker between the users and the advertising providers. The main advantages of the proposed alternative advertisement model are as follows for each of the involved parties:

- Users - can receive target advertisement without disclosing sensitive data (only the predictions are shared) in a transparent way. There might be an incentive to the user from the telco company if the user's visits to places are highly predictable.
- Telco - the telcos can push relevant content, such as SMS advertisements, that is highly relevant to the mobile users' spatio-temporal context and thus offer a better service to the advertisement providers. In addition, can send more relevant SMS advertisements, if they know where users might move next.

– Advertising providers - offers an additional advertisement channel (other than mobile applications and websites) with significantly higher click-through or conversion rate for their user targeting.

As a practical and novel application of *NextLocation*, we plan to explore this idea further and discuss it with telco providers for feedback from the industry standpoint.

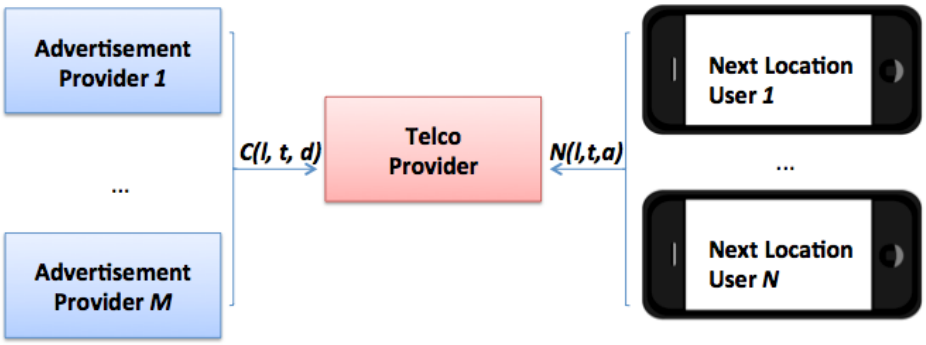

Fig. 3: *NextLocation*: advertisement model

8 Conclusions and Future Work

In this paper we propose the *NextLocation* framework, that is a mobile data mining approach to the next place prediction problem. The main advantage of *NextLocation* is that it is a privacy-preserving solution that fully runs on the mobile device itself. Sensitive data about the user locations and context are not disclosed. Moreover, *NextLocation* uses an adaptive anytime model which enables adaptation to changes in the user mobility patterns. Finally, it keeps an estimate of the anytime model accuracy in real-time.

This paper also reports on our experiments analysing data from the Nokia Mobile Data Challenge (MDC). The results on MDC data show great variability in predictive accuracy across users, where irregular users are very difficult to predict while for more regular users it is possible to achieve more than 80% accuracy. To the best of our knowledge, our results achieve the highest predictive accuracy when compared with existing approaches. Furthermore, we proposed an alternative mobile advertising model that can be implemented using *NextLocation*.

In future work, in line with the last experiments on online learning conducted in this work we plan to develop an online algorithm particularly designed for next place prediction. We are also in contact with telcos to discuss the implementation of the proposed alternative advertisement model.

Acknowledgements

We thank the Nokia MDC organisers for providing their data for research and publication. Our gratitude also goes out to Cao Hong, Minh Nhut Nguyen and Xiaoli Li for working on the data with us.

References

1. P. Barwise and C. Strong. Permission-based mobile advertising. *Journal of interactive Marketing*, 16(1):14–24, 2002.
2. A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer. Moa: Massive online analysis. *The Journal of Machine Learning Research*, 11:1601–1604, 2010.
3. V. Etter, M. Kafsi, and E. Kazemi. Been there, done that: What your mobility traces reveal about your behavior. 2012.
4. V. Fox, J. Hightower, L. Liao, D. Schulz, and G. Borriello. Bayesian filtering for location estimation. *Pervasive Computing, IEEE*, 2(3):24–33, 2003.
5. J. Gama, P. Medas, G. Castillo, and P. Rodrigues. Learning with drift detection. *Advances in Artificial Intelligence–SBIA 2004*, pages 66–112, 2004.
6. J. Gama, R. Sebastiao, and P. P. Rodrigues. Issues in evaluation of stream learning algorithms. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 329–338. ACM New York, NY, USA, 2009.
7. R. K. Ganti, F. Ye, and H. Lei. Mobile crowdsensing: current state and future challenges. *IEEE Communications Magazine*, 49(11):32–39, 2011.
8. H. Gao, J. Tang, and H. Liu. Mobile location prediction in spatio-temporal context.
9. J. B. Gomes, S. Krishnaswamy, M. Gaber, P. Sousa, and E. Ruiz. Mars: A personalised mobile activity recognition system. In *MDM*, pages 316–319. IEEE Computer Society, 2012.
10. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.
11. J. K. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T. M. T. Do, O. Dousse, J. Eberle, and M. Miettinen. The mobile data challenge: Big data for mobile computing research. In *Mobile Data Challenge by Nokia Workshop, in conjunction with Int. Conf. on Pervasive Computing, Newcastle, UK*, 2012.
12. Z. Lu, Y. Zhu, V. W. Zheng, and Q. Yang. Next place prediction by learning with multiple models.
13. B. Ly. Mobile data challenge 2012: Unlocking the secrets of smartphone data, 2012. [Online; accessed 3-December-2012].
14. W. Mathew, R. Raposo, and B. Martins. Predicting future locations with hidden markov models. 2012.
15. A. Monreale, F. Pinelli, R. Trasarti, and F. Giannotti. Wherenext: a location predictor on trajectory pattern mining. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 637–646. ACM, 2009.
16. A. T. Palma, V. Bogorny, B. Kuijpers, and L. O. Alvares. A clustering-based approach for discovering interesting places in trajectories. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 863–868. ACM, 2008.
17. R. Pan, J. Zhao, V. W. Zheng, J. J. Pan, D. Shen, S. J. Pan, and Q. Yang. Domain-constrained semi-supervised mining of tracking models in sensor networks. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1023–1027. ACM, 2007.
18. W. Sherchan, P. P. Jayaraman, S. Krishnaswamy, A. Zaslavsky, S. Loke, and A. Sinha. Using on-the-move mining for mobile crowdsensing. In *Mobile Data Management (MDM), 2012 IEEE 13th International Conference on*, pages 115–124. IEEE, 2012.
19. S. Spaccapietra, C. Parent, M. L. Damiani, J. A. De Macedo, F. Porto, and C. Vangenot. A conceptual view on trajectories. *Data & knowledge engineering*, 65(1):126–146, 2008.
20. L. H. Tran, M. Catasta, L. K. McDowell, and K. Aberer. Next place prediction using mobile data.
21. J. Wang and B. Prabhala. Periodicity based next place prediction.
22. G. Yavaş, D. Katsaros, Ö. Ulusoy, and Y. Manolopoulos. A data mining approach for location prediction in mobile environments. *Data & Knowledge Engineering*, 54(2):121–146, 2005.
23. M. Zook, M. Graham, T. Shelton, and S. Gorman. Volunteered geographic information and crowdsourcing disaster relief: a case study of the haitian earthquake. *World Medical & Health Policy*, 2(2):7–33, 2012.