Comprehensive Predictions of Tourists' Next Visit Location Based on Call Detail Records using Machine Learning and Deep Learning methods

Nai Chun Chen
Massachusetts Institute of Technology
77 Mass Ave
Cambridge, MA, USA
+1 617 253 8799
naichun@mit.edu

Wanqin Xie
Massachusetts Institute of Technology
77 Mass Ave
Cambridge, MA, USA
+1 617 253 8483
hwx@mit.edu

Jenny Xie
Wellesley College
21 Wellesley College Road
Wellesley, MA, MA
+1 617 253 8799
jxie2@wellesley.edu

Kent Larson
Massachusetts Institute of Technology
77 Mass Ave
Cambridge, MA, USA
+1 617 253 8799
kll@mit.edu

Roy E. Welsch
Massachusetts Institute of Technology
77 Mass Ave
Cambridge, MA, USA
+1 617 253 6601
rwelsch@mit.edu

ABSTRACT
Recent developments in data mining and machine learning have helped to solve many issues in prediction and recommendation. In this project, we run a comprehensive study on individual behavior patterns from call detail records (CDR) data to predict tourists' future stops. Multiple classification algorithms are employed, including Decision Tree, Random Forest, Neural Network, Naïve Bayes and SVM. In addition, a Recurrent Neural Network-Long Short Term Memory (LSTM) that is ordinarily applied to language inference problems is tested. Surprisingly, we find that LSTM provides us with the best prediction (94.8%), while Random Forest/Neural Network give the second best (85%). Our investigation suggests that the memory-dependence property of LSTM architecture gives it great expressive power to model our time-series location data, making it an outstanding classifier.

CCS Concepts
• Information System →Information System Application→ Data Mining

Keywords
Data Mining; Call Detail Record; Next Location Prediction; Deep Learning

1. INTRODUCTION

1.1 Overview and Motivation
The Developments in data mining and deep learning have enabled more complex studies that have resulted in meaningful advances in pattern recognition, classification and prediction [1]. One application that has benefitted from this work are recommendation systems. For example, both Google and Yelp Nearby are currently attempting to provide personalized recommendations related to many products including food and shopping. For Andorra, a tourism-centered country, the ability to predict tourist behaviors and destinations is crucial to the national economy. In particular, local business owners gain significant benefits from knowing who their potential customers are and the movement patterns of their customers. We aim to provide recommendations to tourists, beginning with the prediction of future travel routines based on a comprehensive set of features for each tourist. Though this project focuses on the case of Andorra, the methods employed here are applicable to similar problems in other cities and countries.

1.2 Related Previous

1.2.1 CDR data processing
Previous work surveyed in recent reviews indicated that CDR data has been applied to various problems including estimation of population size, residents' urban movement patterns, local events and migration patterns [2, 3, 4]. The most related case is Nokia's Mobile Data Challenge where Do & Garcica-Perez [5] used behavioral patterns from smartphone *apps* data, having much more sequential information, to study how much improvement generic behavioral information can make towards the predictive performance of personalized models. Nevertheless, most of those papers focus only on residents' routine lives.

So far, none of the above work has, to our knowledge, applied comprehensive human behavioral features to predict tourists' large scale, city-to-city movements. Residents' routine daily travel, with at most 3 different stops per day, may be more repetitive in nature [6, 7]. Tourists could be more "exploratory" in their movements. In this experiment, the majority of tourists visited five out of six cities in Andorra. Therefore, tourism data can require a different set of analytic tools to study.

We aim to see if it is possible to capture and anticipate these travel patterns from tourists' previous CDR data.

978-1-5386-1996-4/17 $31.00 © 2017 IEEE
DOI 10.1109/BigDataCongress.2017.10

1.2.2 Related prediction algorithms

Work on predicting next location has been motivated by its potential to create personalized recommendations and context-aware services. Depending on the context, researchers choose how to incorporate different features and different models. Markov models are very popular, and have been used in [8, 9] to analyze GPS data that are much denser compared to CDR data. Krumm et al's paper on inferring destinations from partial trajectories depending on Bayesian inference [10]. In Do and Garcia-Perez's paper using apps data, researchers showed that incorporating multiple features improved the accuracy of predicting human behaviors when applied to both random forest and linear regression models [5]. Nguyen et al's paper shows how these different models can be combined to create fusion classification algorithms [11].

Recurrent neural networks (RNN) have not been as popularized as Markov and Bayesian models for the task of predicting next location. Application of RNN in predicting user behavior has been recently investigated in the context of ad-clicks by Zhang et al [12]. Liu et al explores how modified simple RNN can be used to predict next location by creating a spatial-temporal recurrent neural network algorithm in his paper [13]. In our paper, because data are sparse and lack exact context for each geolocation that should be entered as the second hidden layer, Liu et al's methodology is not very helpful for this problem.

In sum, not much work has been done on predicting the next location of tourists based on CDR using machine learning and deep learning methods, we aim to compare various traditional AI algorithms with machine learning and deep learning algorithms to see which one is better.

1.3 Tourism in Andorra

Andorra is a popular tourist destination in Europe. It is a small country bordered by Spain and France, 80% of whose national GDP comes from tourism [6, 7]. Over the past year, the office of Andorran tourism has noticed a decline in the amount of tourism through their country. Is it possible to predict a tourist's future travel plans to facilitate tourism in Andorra? More specifically, we want to predict tourists' future stops, based on their prior activities and movement patterns. A reliable prediction model could help improve recommendations and information services to assist their travel planning and to improve their experience. The approach is to frame a tourist's next travel destination as a classification problem by inputting the tourist's country of origin and previous travel behaviors as prediction features. There are

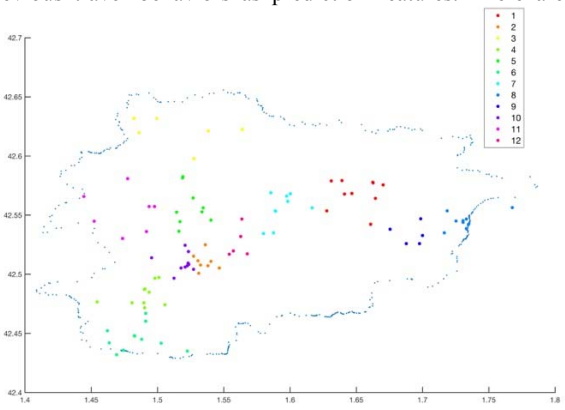

Fig. 1. Cell Tower Map Cluster

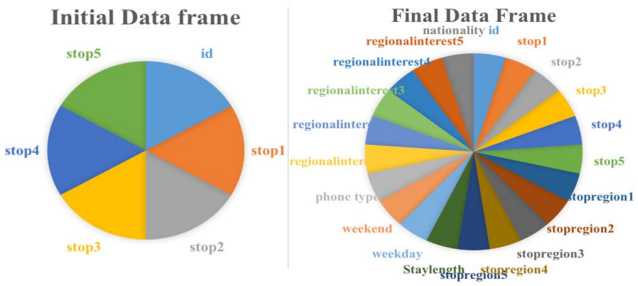

Fig. 2. Features for Initial data frame (left) and Final data frame (right).

numerous methods of classification, each with its own merits, so we sought to test these common types of classification algorithms (Support Vector Machine (SVM), Naive Bayes, Neural Network, Decision Tree, Random Forest) [1] and deep learning methods, such as various types of RNNs, to determine which algorithm would be the most useful for predicting a user's next location.

2. DATA and METHODS

From a total of 16,568,179 data points in January 2015, the average stay length for a tourist was 2.36 ±0.004 days with an average of 4.6 ±0.03 different stops. Hence, we would like to focus on the 5th stop of tourists. Even though Andorra has only 6 districts, the cell towers are divided into 12 clusters with two clusters for downtowns or populated areas as shown in Fig. 1. After running a gap statistic that showed what the number of clusters should be over 9 stops, K-means clustering with k=12 was applied as it has 10 cities in which the most popular one is given an extra cluster and one more for cell towers in foreign neighbor countries. All unknown or missing data were placed in the 13th cluster.

2.1 Original Data

Each row of the original CDR (Call Detail Record) Data contains the following information:

DS_CDNUMORIGEN: cell phone number

DT_CDDATAINICI and DT_CDDATAFI: start time and end time of a call

NUM_DURADA: call length

ID_CELLA_INI and ID_CELLA_FI: start tower and end tower of a call

ID_CDOPERADORORIGEN: cell phone carriers

TAC_IMEI: cell phone type

By analyzing the data above, the goal is to figure out the best prediction performance. Issues with the raw data:

2.1.1 Limited information

The raw data only shows basic user information (phone carrier and phone type), timestamp of phone call, and geo-location of each call. This amount of information is a lot less than needed. In order to create more useful features, we conducted feature engineering on the raw data by adding more features such as regional POIs (Points of interest).

2.1.2 Raw data is sparse

Because of poor signals or the receiver being located outside of Andorra, the cell tower for call end might not be in the records. For those missing cell towers, if the call length is not longer than 300 seconds, then the initial cell tower is assigned as the ending cell tower as well. If the call length is longer than 300 seconds, then an extra category of -1 is created to fill this gap.

2.1.3 Algorithms may require special input

Although the average *number of* geo-location *city visits* is around five (Andorra has six major cities), not every tourist has cell records for the first 5 stops. For methods such as SVM, emptiness is not preferred. Hence, the last recorded stop is used to refill the emptiness, if any. For example, *if User 1* had been to *stop*1, *stop*2 and *stop*3 *only*, then his *stop*4 and *stop*5 will be adjusted to his final stop in appearance*: s*top3.

2.2 Feature Creation and Selection

Our new features are collected in a series of data frames as shown in Fig. 2. The initial frame is created by grouping users and mapping the cell tower geolocation to cities.

Nationality: The nation column is mapped from the phone carrier company to the country.

Weekend/weekday: Two columns are calculated from the timestamps of calls.

StayLength: The StayLength column represents how many days the user visited in a month. This metric is calculated as the difference between the initial time of the first and last calls made by a user.

PhoneType: The PhoneType column is initially categorized as iPhone or non-iPhone and then further details are inferred such as the manufacturer (e.g., Apple, Samsung, Nokia).

RegionInterest: The RegionInterest columns are created using points of interest (POI) located close to the towers. The points of interest are categorized as nature, leisure, events, culture, shopping, gastronomic, wellness, and others.

3. METHODS and RESULTS

Five algorithms were tested for their predictive powers using the input features from the two data frames defined above. Except for Naive Bayes, all other methods (SVM, Decision Tree, Random Forest and Neural Network) give prediction accuracy around 85%, along with higher cross validation scores. In addition, Random Forest and Decision Tree were the fastest methods, compared to SVM and Neural Network.

Two data frames were tested, one with 6 features and one with 21 ones. Random Forest has been applied to the simple data frame, since overfitting was initially a serious issue. If only the first n^{th} stops were used to predict the $(n+1)^{st}$ stop, overfitting will always exist. This problem is inevitable, given that thousand of rows of data indicate so many different choices for the $(n+1)^{st}$ stop.

However, no matter how we change the number of max features and tree listed in the forest, there is always a huge difference between the training and test scores. The training score is often around 100%, while

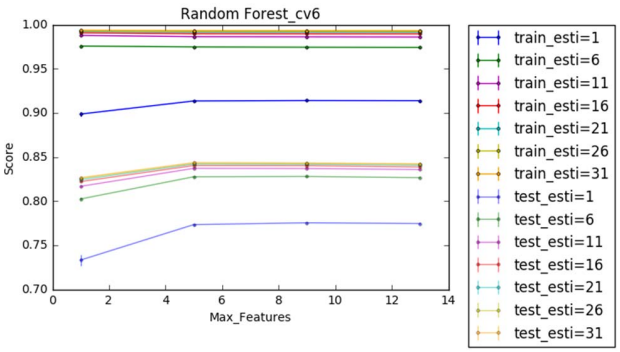

Fig. 3. Server over-fitting within the initial data frame.

none of the test scores exceeded 80%. There exits a very severe overfitting that cannot be improved unless choices for the n+1 stops can be narrowed down [Fig. 3]. Hence, a second data frame with a total of 21 features, including user profile features and spatial features is tested.

3.1 SVM Classifier

We next tried SVM with RBF kernel, and compared the algorithm with different data set sizes. SVM deals well with convex data and can achieve a global optimum with this kind of data, so we were motivated to try it. It is also memory efficient, and a kernel function can be specified for the decision function [8, 14].

In this work, the SVM setup was as follows, retrieved from the scikit-learn library [8]:

For data inputs $x_1, x_2 \ldots x_n \in \{-1,1\}$ and labels $y_1, y_2 \ldots y_n \in \{-1,1\}$,

The primal form of the problem is:

$$\min_{w,b,\delta} \frac{w^T w}{2} + C \sum_{i=1}^{n} \delta_i,$$

$$\forall \, i, y_i(w^T \emptyset(x_i) + b) \geq 1 - \delta_i \; and \; \delta_i \geq 0$$

The dual form of the problem is:

$$\min_{\alpha} \frac{\alpha^T Q \alpha}{2} - e^T \alpha,$$

$$\forall \, i, y^T a \geq and \; 0 \leq \alpha_i \leq C$$

where e is the vector filled with 1's and Q is an $n \times n$ positive semi-definite matrix with $Qij = yiyK(x_i, x_j)$.

The decision function is:

$$sign\left(\sum_{i}^{n} y_i y_j K(x_i, x_j)\right)$$

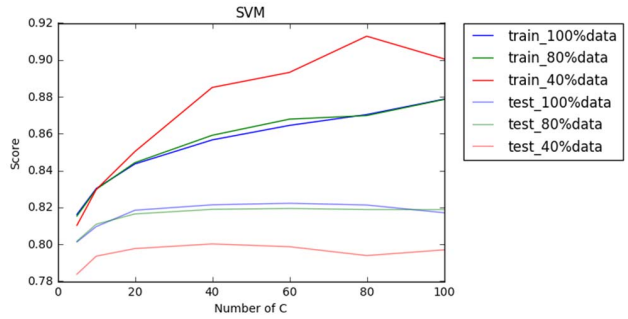

Fig. 4. Accuracy with different C and size of DataFrame

SVMs are powerful tools, but we found their application to be inappropriate here for two reasons: i) the non-convexity of our data made them less accurate than hoped, and ii) the computational complexity and storage of our data made them very expensive and time consuming. The original data set with SVM took 8hr to run. With a smaller data set size, the accuracy moved from 0.82 to 0.79, and it also took 2.5hr to run [Fig. 4].

3.2 Random Forest and Decision Trees

Compared to SVM, decision trees are more flexible and easily adjustable. However, a single tree in the forest can often cause bias and overfitting. If data contains categorical values, those values with more levels will be included in the tree and those with fewer levels will be neglected, causing bias and overfitting. Random forest prevents these issues. In the forest, trees with randomly selected variables (bootstrap) are formed and compared to avoid potential issues [1, 14]

It would still be very helpful if the maximum depth and features of a tree can be determined before the forest grows. Calculations with 5-fold cross validation show that the best depth is around 11 levels and 8 features. As the number of levels approaches 13 or the number of features exceeds 8, the decision tree starts to over-fit the training data [Fig. 7].

We can see that as the number of maximum features in a tree increases, more complexity causes both training and testing accuracy to increase sharply at first. As the maximum number of features exceeds 8, training scores keep increasing while the testing scores approach a limit. The later behavior indicate that the trees are starting to overfitting the data, as the difference between training and testing scores is growing. [Fig. 6 and Fig. 7]

Therefore, we bound the forest growth to a maximum depth of 11 and number of features to a maximum of 8 for individual trees. Having optimized the number of features in each tree, the other major parameter of Random Forests is the number of trees (estimators) in each forest. Number of trees (estimators) can help avoid over-fitting, as having more trees in the forest will balance the importance of the existing features.

The best score obtained is around 85.3% accuracy, once the number of estimators is over 36 [Fig. 8]. Area 4 and area 9 have the most misclassification cases, as these two areas are spatial neighbors in Andorra. The same is true for Area 4 and Area 2, and also for Area 4 and Area 11. Overall, the misclassification rate was at 15%

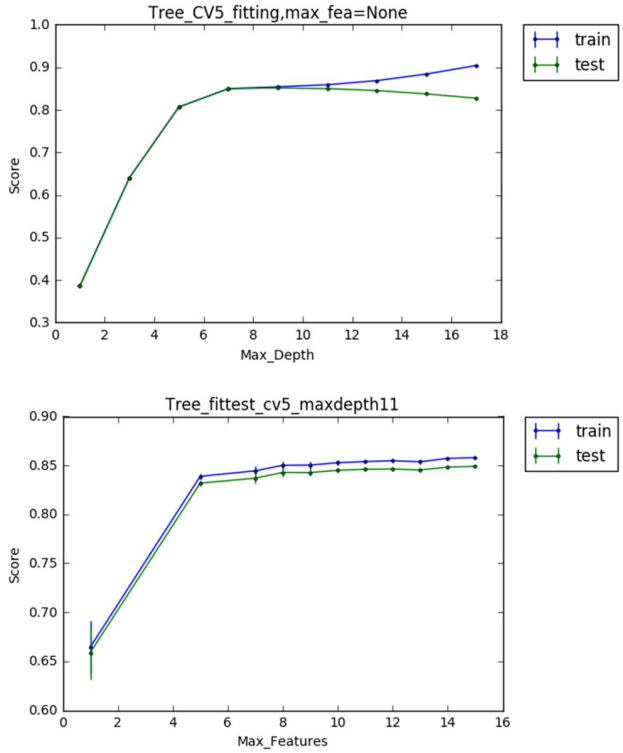

Fig. 6 (up) and Fig. 7(below). Random Forest score VS. Max Depth and number of estimators for one forest sample

3.3 Neural Networks

For classification, neural network is also a good choice. However, the data needs to be standardized for a better analysis. [10,14] The best combination for pattern recognition in the MATLAB neural network toolbox is scaled conjugate gradient (SCG) as a back propagation training algorithm, 'mapmaxmin' as its normalization and Log Sigmoid as its connection function [9, 11, 12].

$$LogSigmoid = \frac{1}{1+e^{n}}$$

Data were split into training, validation/test sets by a ratio of 70/15/15 percent. Two layers were used, one for hidden and one for output. The maximum iteration allowed for each training was

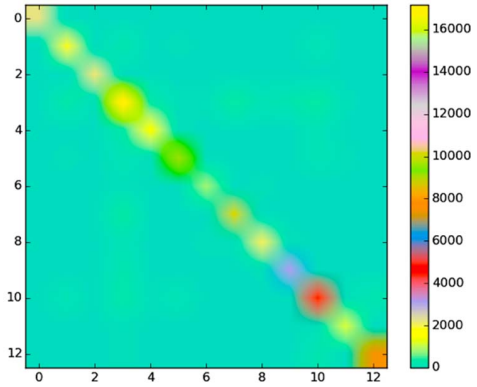

Fig. 8. Confusion matrix from Random Forest

1000. Training would automatically stop if there were no more improvements, indicated by an increase in cross-entropy with 6 fold-cross validations as a default minimum validation time. Different numbers of neurons were tested. There are multiple standards for the number of neurons. Though a popular theoretical number of neurons is between the number of inputs and outputs, broad choices were tested to track down any possible under- or over-fitting [13].

This combination should give the best performance as it filters away weak, highly-correlated features while keeping strong features. Not surprisingly, the best prediction rate is about 85%. The graph clearly shows the generalization accuracy is converging around 85%, as the number of neurons is increasing [Fig. 9]. Hidden layers with 300 neurons were also tested and the training score went up to nearly perfect while the generalization score is still around 85%.

3.4 Naive Bayes

Naive Bayes was also conducted as a supplementary testing method [8, 14]. Unfortunately, due to limitations of the data itself, the prediction score is low, around 0.27. The potential problems within the data are that when a tourist has been to *stop i*, that stop depends on all the stops where he or she has been previously, rather than just the last stop. High dependency between stops requires classical Bayes, instead of Naïve Bayes.

3.5 Recurrent Neural Network

Recurrent Neural Networks (RNN) differs from conventional machine learning algorithms, since it does not process inputs as fixed inputs, but as sequences where states of previous time steps influence the processing of the current time steps. This is useful in the case of predicting the next location, because people travel in a similar manner and their location records are in sequence. The data to be trained in RNN is, therefore, suitable to detect the next location for this type of sparse sequence.

Given that the order of locations should help predict which city the tourists may visit next, using dropout with RNN is also a powerful regularization method that can prevent overfitting. The dropout rate was set to be 0.2. Considering time efficiency, only one hidden layer was used. Data are normalized using 'Maxmin' rescaling.

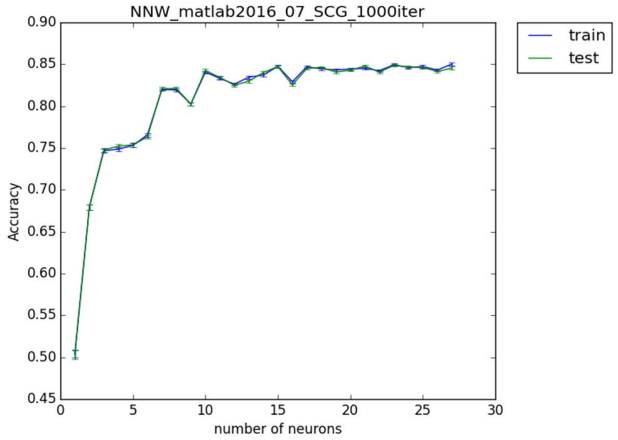

Fig. 9. Neural Network accuracy

In the experiment, both RNN with gated recurrent units (GRUs) and Long short term memory (LSTM) were tried. In all cases RNN proves to be more powerful than conventional algorithms, reflecting the importance of sequences in predicting next location. As an example, RNN-LSTM with all user features can converge at an accuracy of 94.8%, which is significantly higher than the conventional algorithms which only has an accuracy of about 85%.

4. CONCLUSION

In this project, performances of several machine learning methods and time involved deep learning methods were compared. We employed Random forest, SVM, Neural Net, Naïve Bayes, RNN-LSTM, RNN-GRU to predict tourists' next stops using existing stops, other related information and POI (point of interest). We found that, overall, time-involved deep learning method RNN-LSTM (Recurrent neural network, long short term memory) performs about 7% in accuracy better than regular machine learning methods in which the Random Forest and Neural Network are relatively better choices. The worst algorithms for our data were Naïve Bayes and SVM, which were both computationally expensive, and limited. That should be due to the nature construction of the data and its time-related properties.

Our investigations suggest that even though, tourists visit more places than residents, their next location are not harder to predict. Based on the predictions from the algorithms, we can provide useful information for governments to make more attractive and informed recommendations to tourists in the future.

5. FUTURE DIRECTIONS

This study initiates the aggregated recommendation system to tourists based on their existing personal information, past locations and related context. It identifies several features of tourist CDR data that are fundamentally important to enable inference. It also identifies algorithms that are particularly well-suited for analyzing this tourism data.

Second, we wish to try Conditional Random Fields on our data. It has the advantage of having some "logic" within its architecture, and we wish to test if this algorithm has very high performance. A positive answer would reaffirm our hypothesis about the logical nature of our tourist data.

Our future work aims to create recommendation systems for tourists based on their travel histories. In Andorra, review sites such as Yelp are unavailable, so predictive services based on our models may be very useful for travelers to that country.

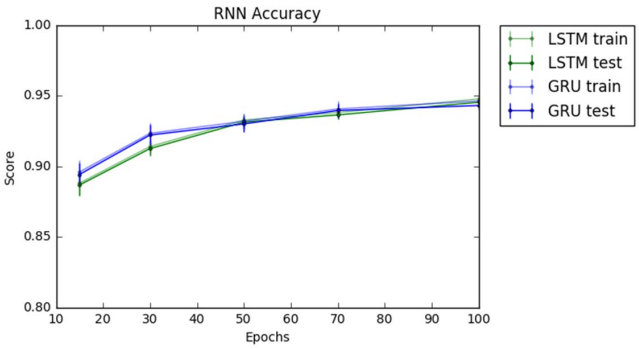

Fig. 10. LSTM Accuracy

6. ACKNOWLEDGMENTS

We thank T.Jaakkola and R.Barzilay for discussions and paper preparation, C. Sun, J. Nawyn, and J. M. Cunningham for helpful discussions and comments, and members of Media Lab Changing Place group. This work was part of the "Applying Machine Learning in Tourism of Andorra" project, which was supported by the Andorra Government.

7. REFERENCES

[1] Alpaydin, E. Introduction to Machine Learning. Cambridge, MA: MIT Press, 2010.

[2] Calabrese, F, Ferrari, L, IBM Research, Ireland, Urban Sensing Using Mobile Phone Network Data: A survey of Research.

[3] Blondel, V, Decuyper, D, Krings, G, Physics. Soc-Ph, A survey of results on mobile phone datasets analysis, Feb, 2015.

[4] Gomes, J, Clifton P, and Shonali K., Data Warehousing and Knowledge Discovery Lecture Notes in Computer Science,146-58, 2013

[5] Lauril J, Gatica-Perez D, Do T, Pervasive and Mobile Computing, Special Issue on the Mobile Data Challenge, August 2013.

[6] Khan F., Ali M., Dev H.: "A Hierarchical Approach for Identifying User Activity Patterns from Mobile Phone Call Detail Records", in *Proc. of IEEE,* 2015.

[7] Liu F., Janssens D., Wets G., Cools M.: "Annotating mobile phone location data with activity purposes using machine learning algorithms", in *Expert Systems with Applications: An International Journal* 40:9, 3299-3311, 2013.

[8] Jaiswal, A.; Chiang, Y.; Knoblock, C. A; and Lan, L. "Location Prediction with Sparse GPS Data" in Proceedings of the 8th International Conference on Geographic Information Science*, 315-219, 2014*

[9] Ashbrook, D.; Starner, T. "Using GPS to learn significant locations and predict movements across multiple users" Personal and Ubiquitous computing *7:5, 275-286. 2003*

[10] Krumm, J.; Horvitz, E. "Predestination: Inferring Destinations form Partial Trajectories". Ubiquitous Computing. Vol 4206. 243-260, 2006.

[11] Nguyen L.T.; Cheng, H.T.; Wu, P.; Buthpitiya, S.; Zhu, J.; Zhang, Y. "PnLUM: System for Prediction of Next Location for Users with Mobility'. In: Proceedings of mobile data challenge by Nokia workshop at the tenth international conference on pervasive computing, 2012.

[12] Zhang, Y.; Dai, H.; Xu, C.; Feng, J.; Wang, T.; Bian, J.; Wang, B.; Liu, T.Y. "Sequential Click Prediction for Sponsored Search with Recurrent Neural Networks". In: Proceedings of the Twenty-Eight AAAI Conference on Artificial Intelligence*, 1369-1375, 2014.*

[13] Liu, W.; Wu, S.; Wang, L,; Tan, T. "Predicting the next location: A recurrent model with spatial and temporal context". In: Thirtieth AAAI Conference on Artificial Intelligence. 194 - 200, 2016.

[14] "Andorra 2016: Best of Andorra Tourism - TripAdvisor." Andorra 2016: Best of Andorra Tourism - TripAdvisor. Accessed May 11, 2016. https://www.tripadvisor.com/Tourism-g190391-Andorra-Vacations.html

[15] The World Factbook, Andorra, CIA Publications

[16] Scikit-learn Tutorials, Scikit-learn 0.17.1 Documentation. Accessed May 11, 2016.

[17] Goodfellow I, Farley D, Mirza M, Maxout Networks, Sept 2013

[18] Boger Z, Guterman. H, 1997, "Knowledge extraction from artificial neural network models," IEEE Systems, Man, and Cybernetics Conference, Orlando, FL, USA

[19] Berry M, Linoff G, 1997, Data Mining Techniques, NY: John Wiley & Sons

[20] Matlab Neural Network Documentation for R2016

[21] Saurabh K, IETT, V3-Issue 6, 2012

[22] Mitchell, Tom M. Machine Learning. New York: McGraw-Hill, 1997