

# Significant Location Detection & Prediction in Cellular Networks Using Artificial Neural Networks

Cristian-Liviu Leca<sup>\*</sup>, Ioan Nicolaescu, Cristian-Iulian Rîncu

Department of Communications and Military Electronic Systems, Faculty of Military Electronic and Information Systems, Military Technical Academy, Romania

Copyright © 2015 Horizon Research Publishing All rights reserved.

**Abstract** Location services and applications, based on network data or global positioning systems, are greatly influencing and changing the way people use mobile phone networks by improving not only user-applications but also the network management part. These applications and services can be further developed by introducing location prediction. We design a system that logs cell id and timestamp data from the users' mobile device, detects the significance of the location to the user, such as home and workplace, and predicts future locations over a chosen time period using artificial neural networks. A novel method is designed for location detection that automatically determines the significance of the location to the user, by spatial and temporal analysis. In our approach, the neural network is automatically adapted, with the help of the location detection algorithm, to the period of the week for which a prediction is desired, achieving accurate weekday and weekend location prediction.

**Keywords** Artificial Neural Network, Location Detection, Prediction, Network Management, Open Cell ID

## 1. Introduction

The widespread availability of mobile communication devices together with the development of location-based services and applications has led to an increased research interest towards efficient methods for the detection and prediction of significant locations where a mobile user will travel. These methods aim to improve both end-user applications, such as route planning, car pooling, meeting planners or location-based advertisements, and also be combined in order to solve network management issues [1,2] such as resource allocation, traffic planning, QoS improvement, availability, etc.

Location prediction can be used as an efficient proactive network management measure. For example, in order to ensure a smooth handover between neighboring cells during an audio call, a common proactive measure is to allocate

resources in all neighboring cells. The use of location prediction offers more precise resource allocation and reduces the level of network signaling by allocating resources only inside the most probable future cell.

In order to achieve location prediction we must assume that the mobile device user follows specific patterns, that can be modeled, between locations of significance such as home, workplace, favorite gym/park, downtown. Location prediction is also influenced by variations in the schedule of a person during a week., caused by different habits on working days and in weekends.

Previous work on location detection and prediction use methods such as statistics, data mining, handover analysis [3], Markov models [4-6], decision trees, artificial neural networks (ANN) [7,8].

Most related work on future location prediction does not focus on location detection and do not try to find the related significance of the location which could be used to enhance the results of the prediction methods. Many of these works rely on extracting spatial trajectories and don't take into consideration the user schedule and the calendar data regarding the desired day for prediction. Also some of the proposed methods only focus on predicting the next-cell, rather than the users movement patterns over a chosen time period as our method proposes.

Xu Chen et al. [9] propose a machine learning prediction system aimed at predicting the next cell id where a mobile device will handover to. Their system uses location data gathered from the cellular network consisting of Channel State Information and handover history. Their proposed method treats location prediction as a classification problem solved by applying Support Vector Machines (SVM). Another proposed method aimed at predicting only the next location is presented in [10], which uses sequence mining by the use of multiple support thresholds for different levels of pattern generation.

Xiaofeng et al. [11] propose a location prediction algorithm that will make use of directional antennas. Their work is mainly oriented at predicting and tracking location of moving vehicles.

Parija et al. [8] employ the use neural networks as a

solution the location management problem in cellular networks. Their work proposes a prediction based location management scheme using the multilayer perceptron as basis.

The work presented in [5] proposes a hybrid method for future location prediction using Hidden Markov Models (HMMs). Their approach clusters location histories according to their characteristics, and latter trains HMMs for each cluster. The HMMs are then used for location prediction. Their method of location clustering has the same objective as our location detection method, though different techniques are employed.

For location prediction to be precise, the identification of the boundaries and the significance of the user locations is needed. Our proposed method for location detection combines spatial and temporal analysis in order to extract the significance of the location to the user. The extracted significance together with weekday data (working day or weekend) is then used to adapt the neural network according to the desired day for location prediction.

In this paper we describe the steps involved in designing a system that can obtain location detection and prediction using mobile phone data. In the second section we present the resources and assumptions that were involved in designing the system. The third section describes the rules behind the proposed algorithm used for detecting significant locations. In the fourth section we describe the neural network used for location prediction. The fifth section present the results obtained in detecting and predicting future locations. The sixth section summarizes our results and draws some possible improvements and future research goals.

## 2. Resources Involved

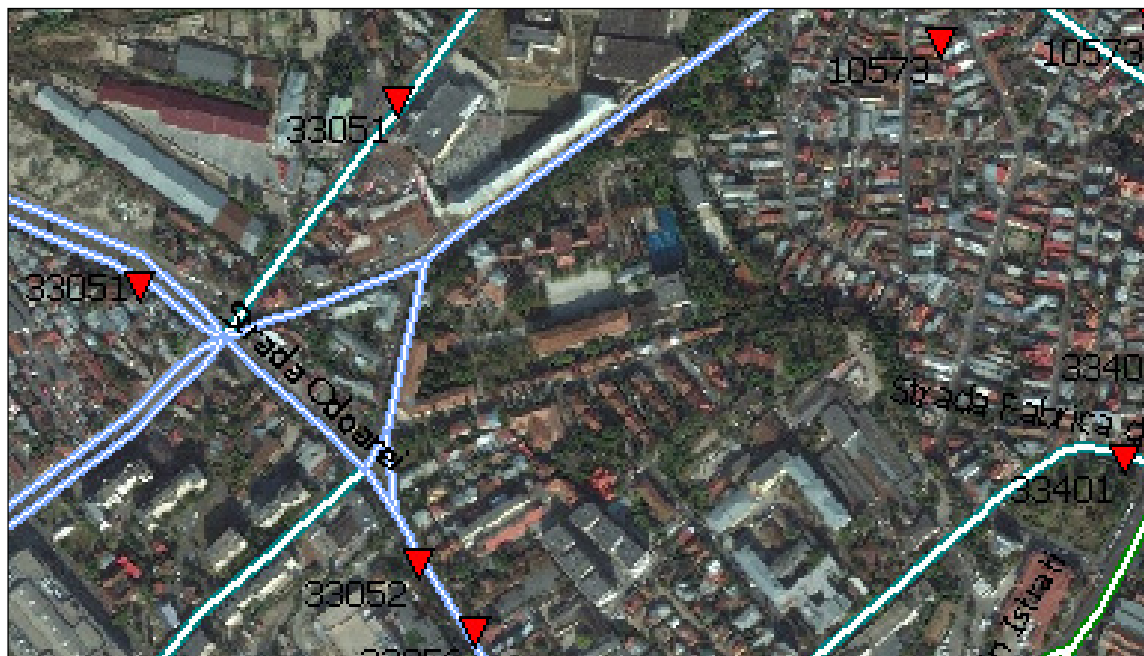
The current location of a mobile device can be obtained in different ways: as exact GPS coordinates (latitude/longitude) or as symbolic coordinates [6]. The symbolic coordinates can be represented by the GSM base stations cell id or the wireless access point name. By using the known GPS coordinates of GSM base stations, the symbolic coordinates represented by the cell id can be linked to real coordinates. Exact coordinates are more useful for end-user applications, while network applications can obtain performance using only symbolic coordinates.

In order to obtain GSM cell tower coordinates for mobile network operators in Romania we have used the free and public available OpenCellid database. The database contains around six million unique cell ids collected from around the world, and supplies information regarding the mobile country code, mobile network code, location area code, cell id, number and date of measurements of the cell id.

Another supplier of cell tower coordinates is the Google Geolocation API, which is limited to 100 interrogations per day in the free version. We preferred to use OpenCellid, as it offers free access to the complete database.

The cell towers were drawn, using the GPS coordinates, on a map constructed using Openstreetmap road vector data and satellite imagery as shown in Fig. 1.

OpenCellid manages to determine an estimated GPS position of the cell towers by averaging cell signal measurements from different collecting devices (smartphones running OpenCellid clients or commercial tracking devices) that report their precise GPS coordinates.



**Figure 1.** BTS location (triangles) of Vodafone Romania around the Military Technical Academy

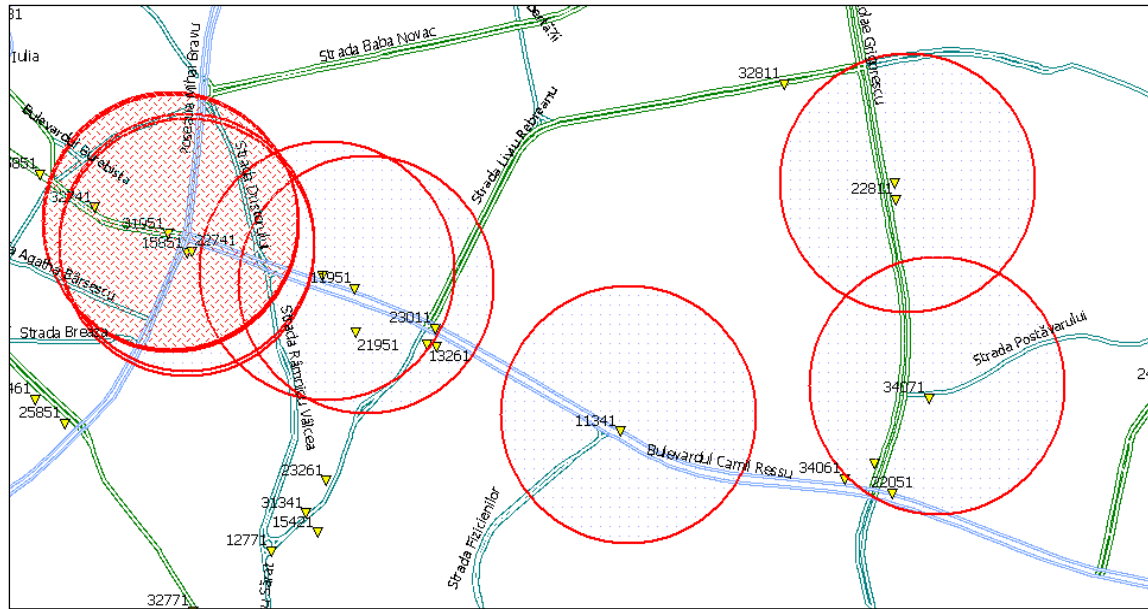


Figure 2. Cell towers mapped around Dristor (Bucharest)

The mobile device cell id information can be obtained using the mobile phone network, as call related information (CRI) and as forced location updates [12], by using radio sensors as proposed in [13] or by using only the mobile device and logging the visited cell ids together with timestamps. We used the second method, sampling and interpolating the data at a half hour interval. Fig. 2 represents the mapping of the cell id (in red circles) and timestamps as in Tab. 1.

Because the OpenCellid database does not supply cell information such as radius, azimuth and directivity, which can be used to further improve location prediction as in [11], we assumed the cells are omnidirectional and we estimated a mean coverage radius for a cell in an urban environment to be around 400 meters, by measuring the medium distance between two cells.

Table 1. Cell id and timestamp example data.

CELL ID	Date	Time
15851	13/12/14	15:15:00
15851	13/12/14	15:45:00
8671	13/12/14	16:15:00
11951	13/12/14	16:45:00
11341	13/12/14	17:15:00
34071	13/12/14	17:45:00
22811	13/12/14	18:15:00
31951	13/12/14	18:45:00
31951	13/12/14	19:15:00

### 3. Location Detection

Different environment conditions such as fading, multi-path propagation, refraction, diffraction, air humidity,

will lead to mobile devices that are being stationary, for example in one room of an apartment, to attach to different cells during a chosen time period. Also short distance walks around one significant place where the user is located could trigger the mobile device to attach to cells further away from the actual location. These facts make it difficult to accurately determine the actual location the person is in, creating the need for a location detection algorithm. The algorithm will integrate all the information and offer a spatial and temporal estimation of the location significance.

We define location detection as the process of extracting the most common locations that are visited by the user. Each detected location will then be analyzed in order to detect its significance to the user, as home place or as workplace.

The goals of the location detection algorithm are: to identify the most common user locations, to convert the symbolic five digit representation of cell ids to a more simplified alphabet based on a smaller number of locations, to improve the accuracy of prediction by feeding a smaller range of values into the prediction algorithm, to detect the significance of the location to each user and to help the prediction algorithm adapt to the day of the week that a prediction is desired.

The proposed algorithm in Fig. 3 first creates a top based on cell ids and the number of occurrences in the analysis time period. It then selects the first occurring cell id and creates a circle object, that will be the first detected location, around the cell. The circular location will be determined by the cell GPS coordinates and a variable radius  $r_a$ , which is set automatically by the algorithm.

In the next step the algorithm intersects the location with all cell objects within its radius and counts the number of cells inside. Next, the algorithm enters a loop fetching another cell from the top. The cell is verified not to be

contained within other locations, if it is the algorithm fetches the next cell. Then a new circular location object with radius  $r_a$ , is created around the cell coordinates. If the current location intersects other previous detected location the  $r_a$  is decremented and the algorithm starts again from the first cell. Fig. 4 represents an example of the algorithm avoid location overlap by decrementing the location radius.

The following notations are used: *decrement radius* is the label of a jump operation, that is called only when two detected locations intersect, *top\_ares* is the table containing the spatial representation of the detected locations, *top\_cells* is the table containing the cells the user visited sorted by the number of occurrences in the analysis period, *area* is the spatial variable containing the currently analyzed location drawn using the radius  $r_a$ , *area\_ctr* is a variable containing the number of records from the analyzed data that are contained inside the current location, *top\_cells\_counter* is a variable depicting the number of occurrences of a cell inside the *top\_cells* table, *current\_top\_cell* is the identifier of the currently analyzed cell id.

```

decrement radius:  $r_a = r_a - 50$ 
initialize  $r_a$  and create table top_areas
fetch first from top_cells
do while not (end of(top_cells))
  fetch current record from top_cells
   $area = createcircle(x,y,r_a)$ 
   $area\_ctr = count(cells\ intersecting\ area)$ 
  if (top_cells_counter = 1 and  $area\_ctr > 1$ )
    fetch next record and go to end loop
  if (top_areas intersects current_top_cell)
    fetch next record and go to end loop
  else
    if (top_areas intersects  $area$ )
      delete table top_areas
      go to decrement radius
    else insert  $area$  into table top_areas
  fetch next record
loop

```

Figure 3. Location detection algorithm

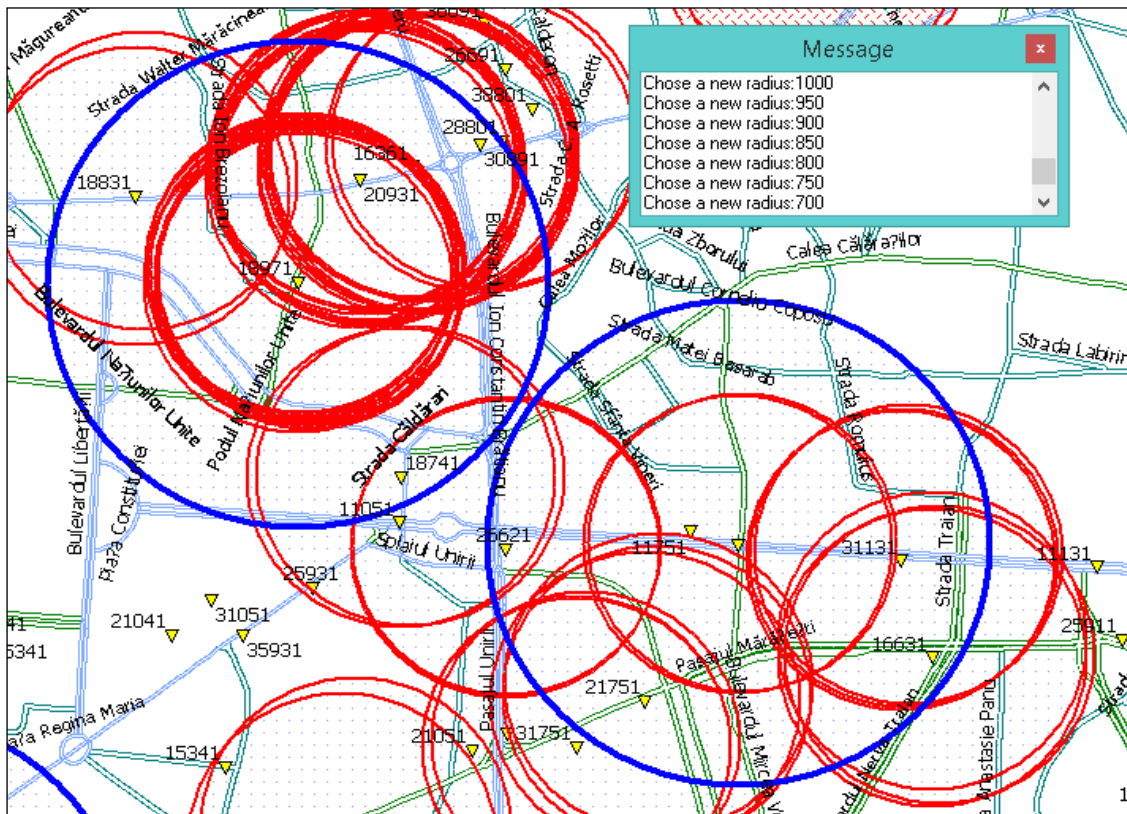


Figure 4. Avoiding location overlap by automatically decreasing the radius

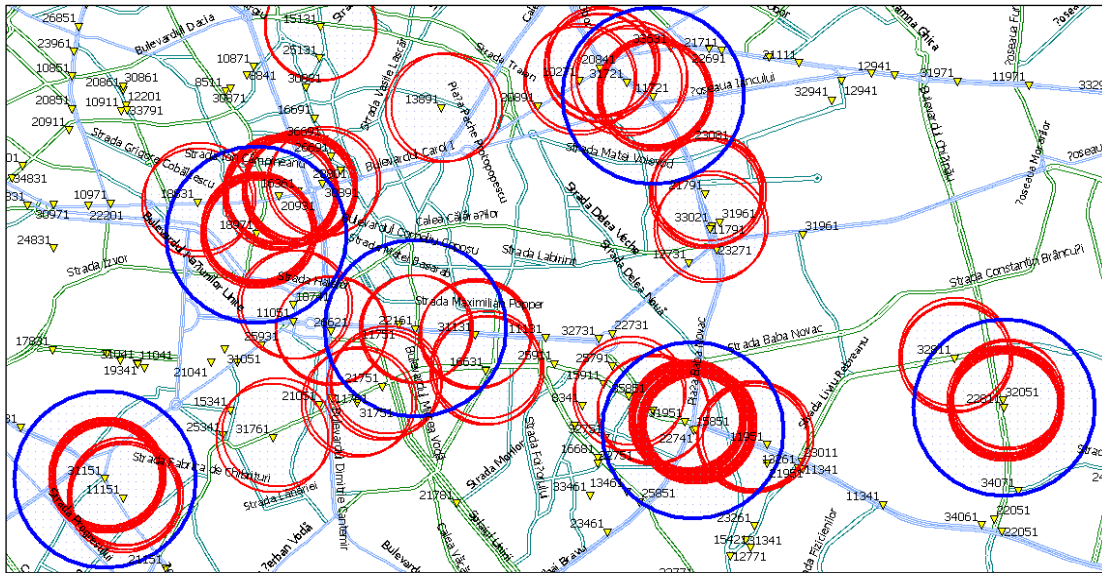


Figure 5. Detected areas for two week data

The algorithm will also discard locations obtained from single occurrences of a cell that intersects a number of other cells creating a false significant location. This is shown in Fig. 5 as cell objects that are not included in any location.

The result of the algorithm will be stored in a table, that contains all detected areas, the id of the originating cell and the total number of records that are contained inside the location.

Furthermore, location detection can be improved not only in spatial context, but also in temporal context, as most people schedule their days between their home place and workplace. This makes it possible not only to detect significant locations for users, but also to learn the significance of the location to the person.

By knowing the schedule that a person follows at his workplace, it is easier to predict the next movement he could make after leaving work. Also it is possible to use different prediction rules for workdays and for weekends and holidays [14], allowing the prediction method to quickly adapt and learn new behaviors in the person’s movement patterns.

In order to detect home place and workplace we completed the resulting significant locations from the previous algorithm with the schedule inside each location by counting the number of occurrences inside the area for all twenty-four hours. Then we applied the set of rules, similar to [15], presented in Tab. 2 for determining the location type. The rules are based on the most common schedule that people in Bucharest use at the workplace.

Table 2. Location type detection rules

Location	Rules
Home	- user is home between 8 pm and 7 am - user visits the location in the defined interval at least 4 days a week
Work	- user is at work between 8 am and 3 pm - user visits the location in the defined interval at least 3.5 days a week

The ways the rules were set allow for small but usual changes in everyday behavior such as taking a day off work, or spending the night dining or partying away from home.

Finally we sort the locations according to the total number of occurrences in each and we assign them an id.

#### 4. Location Prediction

Once the large amount of location data obtained from the user movement patterns has been reduced to significant locations, the next step is to process the resulting data, in order to permit further operations that should not be influenced by the numerical representation of the cell id. For that we replace the cell id with the location id, obtaining a simple model of the user movements as a discrete time series.

From the available techniques for obtaining time series prediction [16] we chose the use of neural networks, because they allow the estimation and approximation of functions that are influenced by a large number of inputs, as a person’s daily route is. This allows accurate predictions of workplace and home place schedule but also the precise moments the person leaves or arrives and the intermediate route he would take.

The method used for training was supervised learning. Supervised learning allows the neural network to infer the function behind the user daily and weekly movements by training with data consisting of the users past movements. The training data will be organized in training examples consisting of pairs formed by an input vector and the desired output value. The training algorithm will analyze the data and the relations between the input and the output and will produce an inferred function. The training algorithm must allow the network to generalize to data that was not fed into the network during training, which translates to the ability of the network to adapt and predict

user movements inside newly visited locations that were not used for training. The volume of training data will be strongly related to the amount of complexity seen in the users daily movements. For individuals which follow a strict routine between home place and workplace, training with a small amount of data consisting of 24 hour movement patterns will allow accurate predictions. In the case of user with highly complex movement patterns that make often visits to a larger number of locations, testing revealed that accurate predictions were possible after training with at least 2 week movement patterns.

We used the nonlinear autoregressive neural network design available in the MATLAB software, as it is suitable for predicting a time series using past values by using supervised learning. The network was trained using Bayesian regularization, together with the Levenberg-Marquardt optimization algorithm, implemented as the *trainbr* function in MATLAB [17,18]. When compared to the other training functions available, the chosen function offered more precise result at the cost of training times as much as ten times longer. Bayesian regularization will aim at minimizing a linear combination of squared errors and neuron weights. The minimization will take into account the need for generalization qualities of the network. The regularization will take place within the Levenberg-Marquardt algorithm. The algorithm is also

known as the damped least-squares method and is a popular curve-fitting algorithm.

The neural network structure and number of hidden layers was determined by repeated trials using test data, with a result of three hidden layers which consist of 48, 24 and 12 neurons, as shown in Fig. 6. The repeated trials proved that the structure of the networks first hidden layer must be equal or a multiple to the number of input samples gathered in 24 hours. The network is trained in open-loop form, and is used in close-loop form for predicting.

The training data consists of integers representing the id of the location while the output data resulting from the trained network will be represented in double format. This means that in order to assign the data to a location id, we need to apply a rounding function, that will act as a decision block for the output of the network.

### 5. Results

The functioning of the proposed method was tested in real-life scenarios by logging the authors mobile devices movements over different periods inside the network of mobile operator Orange Romania. As mentioned the logs consist of the visited cell ids and the timestamp of the events.

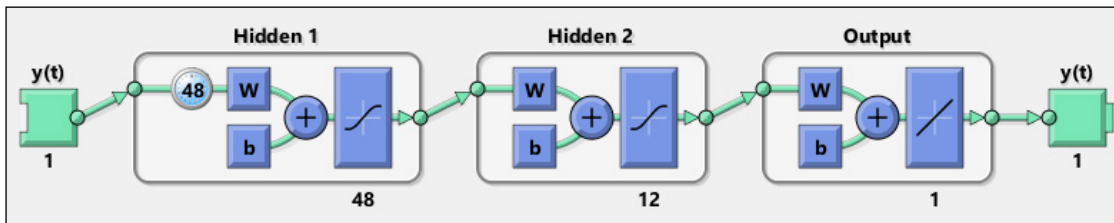


Figure 6. Neural Network Structure in training mode

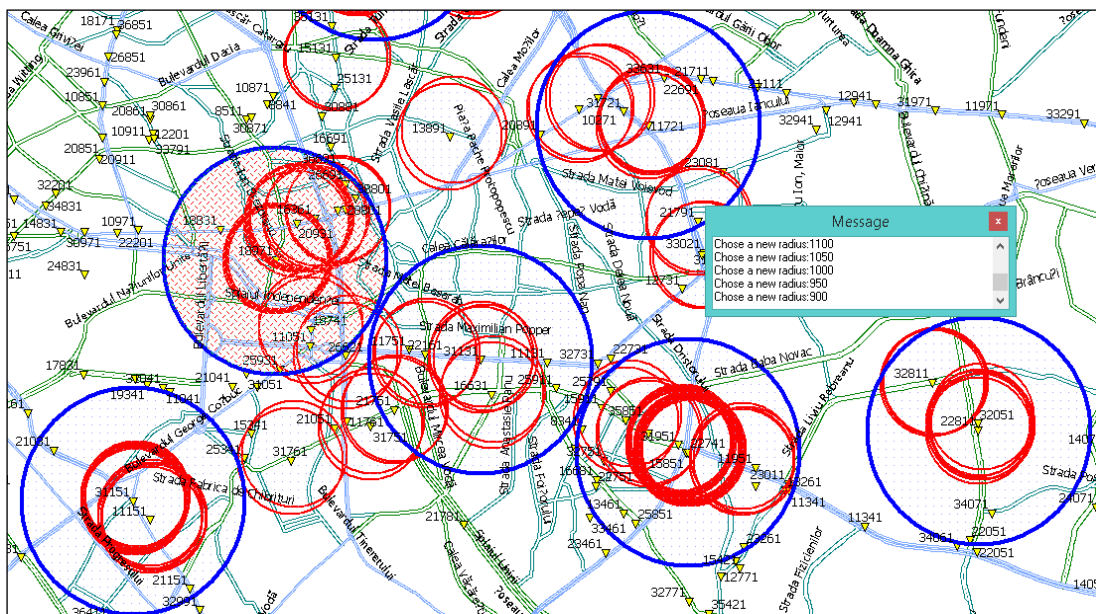


Figure 7. Detected areas for one week data

Simulations proved that the algorithm succeeds in detecting locations and also sorts them according to their importance for data logs with a minimum time span of 24 hours, which allows the detection rules in Tab. 2 to work. The locations are detected correctly, with the help of the half hour sampling interval, but it is important to take into consideration if the data was captured during weekdays, weekends or holidays, which would result in detecting locations that were significant only during that period. For accurate results in location detection and for allowing precise prediction a proper time span should be consisted of at least two weeks data. This usually allows weekday prediction with great accuracy, and finding the usual schedule for the home place in weekends.

Another benefit of longer time spans for the input data is the reduction in radius of the locations, which is caused by repeated visits to the same cells placed on common routes such as between the home place and the workplace. These cells will be detected as significant locations, also allowing for the prediction of the route the user will take between two locations.

A different result of automatic radius reduction is that more cells that were visited only once and don't add up to the significance of locations, will be discarded. This leads to increased precision in finding the exact cell towers that need to be addressed by the network management part.

When analyzing Fig. 7 (obtained using only the first week of the two-week data presented in Fig. 4), the increased precision in detecting the location is observable as the area of the locations in Fig. 4 is smaller by 40%.

The primary disadvantage of detecting more locations will be the increased dynamic range that the neural network has to process. This will make it harder for the neural network to predict locations that are less visited.

The data that we chose to present the prediction results and performances consists of a sixteen day cell id and timestamp log of the authors mobile device, resulting in 768 records. The location detection algorithm managed to detect seven locations, of which five were places that had real significance to the author, and two which were in fact

transit areas between locations as shown in Tab. 3.

**Table 3.** Detected locations

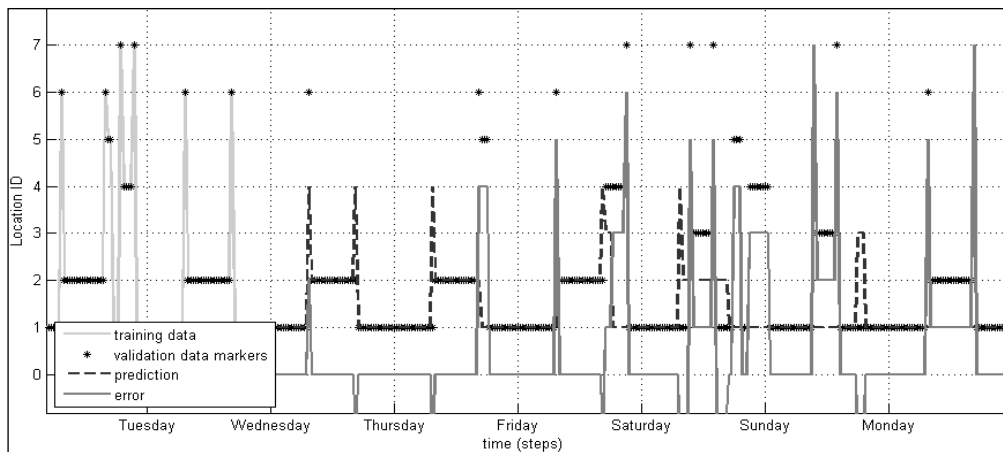
Location ID	Records	Area Type	Real significance
1	452	Home	Home
2	202	Work	Work
3	34	-	University
4	33	-	Downtown
5	15	-	Favorite park
6	19	-	Home to work route
7	12	-	Home to university & downtown route

Cell ids that were not contained within the significant locations were labeled with 8, as location id.

The sixteen day logs were split into training data and validation data. As the number of detected locations was relatively small, training with half the data and using the rest for validation provided accurate prediction.

We noticed that using as less as two days weekdays for training, we obtain accurate prediction for the following three weekdays. But as the weekend interval was not fed into the neural network training, the network fails to predict the time the person leaves his home place and the destination location during the weekend. The network will use a 48 sample delay on the loop, meaning that the network predicts recursively based on the previous day output.

Fig. 8 presents the predicted data from Wednesday to Sunday in black dashed line, the validation data as black star markers, and the error in gray line. The error values are null when the validation data and the prediction overlap, otherwise the differences are visible. Errors generated during the weekdays are mainly caused by failure of the neural network to adapt to the large variations in the location id over the course of one or two time steps. As expected, due to the loop delay, the Saturday prediction is influenced by the previous day output, resulting in an inaccurate prediction of the user going to his workplace.



**Figure 8.** Five day prediction result using two weekday training data

Training with a larger data interval will only increase the prediction accuracy for the weekdays, while the errors generated for the weekend will remain the same as shown in Fig. 9 which was obtained with eight days training data.

For solving this problem we adapt the neural network in order to make use of the spatial and temporal information obtained using the location detection algorithm. The method works by applying different delays to the neural network loop, according to the desired day for prediction. Fig. 10 presents the network adapted for weekend prediction with a

seven day delay.

By applying a seven day delay to the neural network only when used for predicting weekend days movements, it can predict the weekend departure and arrival time from the home place with more accuracy, and also the destination location of the person as shown in Fig. 11.

The neural network manages to predict the departure time for both weekend days and the time at which the person returns home, together with the schedule in the location with id three.

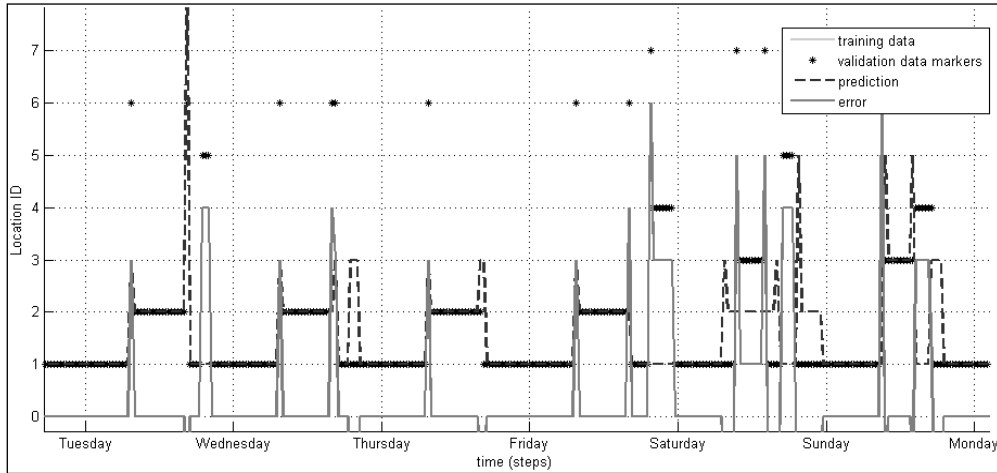


Figure 9. Weekday prediction using eight day training data

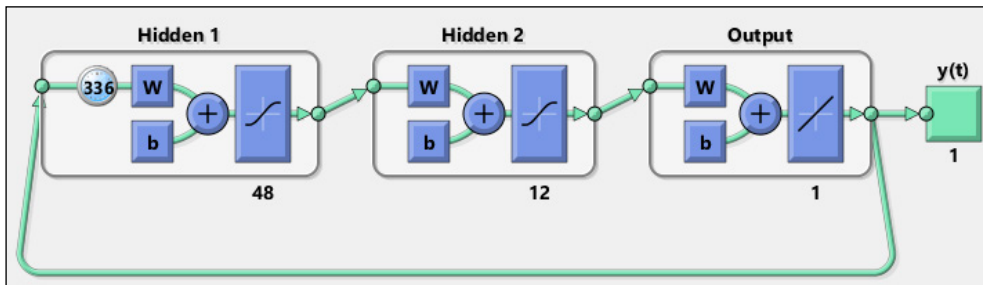


Figure 10. Network structure adapted to weekend prediction

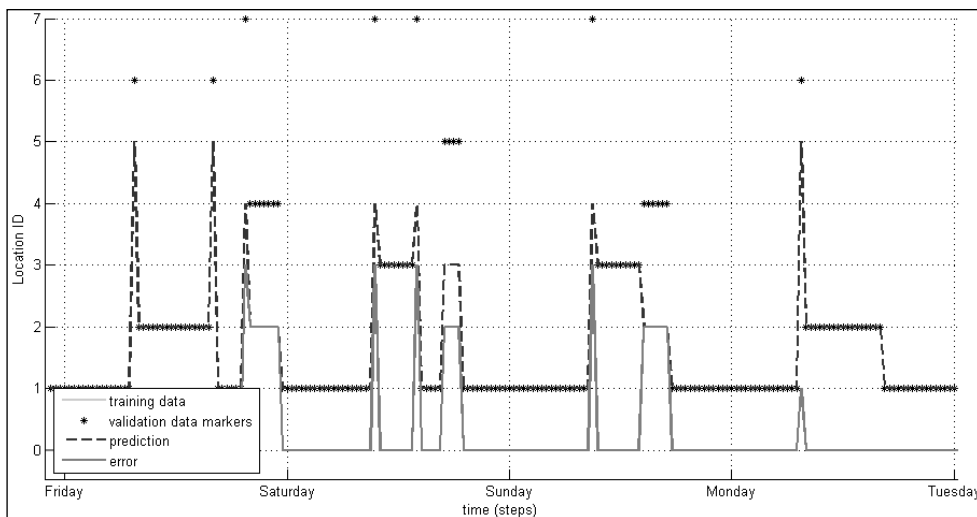


Figure 11. Weekend prediction using different loop delay



## 6. Conclusions & Further Work

This article presents the steps involved in designing a location detection and prediction systems for use in mobile networks, and possible applications for it. The system uses public available data for geospatial representation of the mobile networks. In the proposed approach we use spatial and temporal operations in order to accurately detect locations and their significance to the mobile users. The aim of the location detection algorithm is to create a simplified model of the user movements which can be fed into the neural network used for prediction.

Our approach offers a novel method for using neural network in location prediction. The main idea behind the method is the automatic adaptation of the structure of the neural network according to the location significance information obtained using our location detection algorithm.

We find that predicting weekday movements for most users can be achieved with a minimum amount of training data consisting of two day records, while weekend predictions need at least two week data for accuracy.

As the OpenCellid cell tower location data provided minimum information regarding the real network configuration, we propose as future work to further improve the location detection algorithm by using information regarding cell tower radius, azimuth, directivity and antenna model in use. By combining these information with the received signal indicator in the mobile device, inside a radio propagation model it is possible to greatly reduce the radius of the detected locations. We also propose to implement different methods for prediction such as decision-tree models or hidden Markov models.

---

## REFERENCES

- [1] Vranova, Zuzana, and Vojtech Ondryhal. "Utilization of Selected Data Mining Methods for Communication Network Analysis." *Radio engineering* 20, no. 2 (2011): 549.
- [2] Feng, Huifang, Chunfeng Liu, Yantai Shu, and Oliver WW Yang. "Location Prediction of Vehicles in VANETs Using A Kalman Filter." *Wireless Personal Communications* (2014): 1-17.
- [3] Fülöp, Péter, Sándor Imre, Sándor Szabó, and Tamás Szálka. "Accurate mobility modeling and location prediction based on pattern analysis of handover series in mobile networks." *Mobile Information Systems* 5, no. 3 (2009): 255-289.
- [4] Wen, Li, Xia Shi-xiong, Liu Feng, and Zhang Lei. "Improving Location Prediction by Exploring Spatial-Temporal-Social Ties." *Mathematical Problems in Engineering* 2014 (2014).
- [5] Mathew, Wesley, Ruben Raposo, and Bruno Martins. "Predicting future locations with hidden Markov models." In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pp. 911-918. ACM, 2012.
- [6] Cheng, Christine, Ravi Jain, and Eric van den Berg. "Location prediction algorithms for mobile wireless systems." In *Wireless internet handbook*, pp. 245-263. CRC Press, Inc., 2003.
- [7] Akoush, Sherif, and Ahmed Sameh. "Mobile user movement prediction using bayesian learning for neural networks." In *Proceedings of the 2007 international conference on Wireless communications and mobile computing*, pp. 191-196. ACM, 2007.
- [8] Parija, Smita, Santosh Kumar Nanda, Prasanna Kumar Sahu, and Sudhansu Sekhar Singh. "Location Prediction of Mobility Management Using Soft Computing Techniques in Cellular Network." *International Journal of Computer Network and Information Security (IJCNIS)* 5, no. 6 (2013): 27.
- [9] Chen, Xu, François Mériaux, and Stefan Valentin. "Predicting a user's next cell with supervised learning based on channel states." In *Signal Processing Advances in Wireless Communications (SPAWC), 2013 IEEE 14th Workshop on*, pp. 36-40. IEEE, 2013.
- [10] Keles, Ilkcan, Mert Ozer, I. Hakki Toroslu, Pinar Karagoz, and Salih Ergut. "Location prediction of mobile phone users using apriori-based sequence mining with multiple support thresholds." (2014).
- [11] Lu, Xiaofeng, Pietro Lio, Pan Hui, and Haifei Jin. "A location prediction algorithm for mobile communications using directional antennas." *International Journal of Distributed Sensor Networks* 2013 (2013).
- [12] 3GPP TS 23.012: Location management procedures, 3GPP, 2014.
- [13] Kozak, Filip, and Premysl Hudec. "Radio Sensor for Monitoring of UMTS Mobile Terminals." *Radio engineering* 23, no. 2 (2014).
- [14] Papandrea, Michela, and Silvia Giordano. "Location prediction and mobility modelling for enhanced localization solution." *Journal of Ambient Intelligence and Humanized Computing* 5, no. 3 (2014): 279-295.
- [15] Tran, Le Hung, Michele Catasta, Lucas Kelsey McDowell, and Karl Aberer. "Next place prediction using mobile data." In *Proceedings of the Mobile Data Challenge Workshop (MDC 2012)*, no. EPFL-CONF-182131. 2012.
- [16] Armstrong, Jon Scott, ed. *Principles of forecasting: a handbook for researchers and practitioners*. Vol. 30. Springer Science & Business Media, 2001.
- [17] MacKay, David JC. "Bayesian interpolation." *Neural computation* 4, no. 3 (1992): 415-447.
- [18] Foresee and Hagan, *Proceedings of the International Joint Conference on Neural Networks*, June, 1997.