Neural Networks for Location Prediction in Mobile Networks

Jens Biesterfeld, Elyes Ennigrou, Klaus Jobmann

Institut für Allgemeine Nachrichtentechnik, Universität Hannover
D-30167 Hannover, Germany

biest@ant.uni-hannover.de

Abstract

Location prediction serves to save capacity on the air interface of mobile radio networks. A selection of neural networks of the feed-forward and feedback type are examined to prove their suitability for this purpose. As first results preferable network structures, input vectors, learning parameters and the simulated prediction probabilities are presented. A comparison with conventional methods shows the advantages and disadvantages of the use of neural networks for motion prediction. The results show that the gain depends on the user profile and the amount of extraordinary movements of the subscriber.

1 Introduction

Mobile networks of the next generations will be designed with smaller cells than today's systems, because of the expected growth in the number of subscribers and physical reasons like higher frequency caused by higher bandwidth. The smaller cells cause smaller location areas by keeping the paging efficiency constant. The lower size of the location areas and the higher number of subscribers lead to rising signalling traffic for the purpose of location management. The location updating information is transmitted over the air interface of the mobile system. The air interface is the bottleneck in a wireless system. To save capacity on the air interface, it will be necessary to reduce the signalling traffic.

One main factor in signalling via the air interface is the location updating of the mobile stations which are not in a call but attached to the system. They do not occupy a traffic channel which can be used to transfer the necessary signalling. To prevent overload on the air interface in future systems, it is necessary to find methods to reduce the traffic caused by location updating. There are some proposals to minimise the signalling traffic on the radio link. They reach from creating dynamic location or paging areas to the idea to use the user behaviour and his traffic characteristics. One way to use the movement data of the subscriber is to predict his future location.

2 Location Prediction

2.1 Conventional Methods

Location prediction means the use of the historical movement patters of a subscriber to calculate his possible future locations. To solve this problem some algorithms are proposed. One method bases on a table with the possible locations and the probability the user is located in there in a deterministic period of time [1]. In the case of mobile terminating call the subscriber is paged in the possible location areas, in the order of falling possibility, until the mobile terminal answers.

A second method stores the historical movement pattern in a database and compares the recent states with these movement tracks in the database to find the one witch samples the actual states. If a pattern matches with some criteria the next state from the stored pattern is used to predict the future location of the subscriber [2].

The first algorithm is only dependent on time because the time is the criteria to determine the table index and the second method is only dependent on states. In reality it can be assumed that the location of a subscriber is dependent on time as well as actual state. Both methods do not predict the right location, if the subscriber shows different movement patterns at different times.

One method to combine the time and location information is to use the prediction characteristics of some neural network types.

2.2 Method using Neural Networks

At the beginning the user has to be observed to store his movement patterns as a function of time m(t). This action is the same as in the mentioned methods, because they also need information about the users movements. The first method to record the location and the probability belonging to it. The second algorithm needs no explicit learning phase, but prediction is not possible until some regular movement patterns are detected and stored in the database.

In the next step the suitable neural network has to be trained (s. chap. 4) with the observed motion pattern m(t) (s. chap. 2.3). After training the network the recall phase is used for prediction. The actual movement and time is used to feed the neural network and to get the output the next location or locations depending on the output vector (s. chap. 3). We expect from this method that it shows better results with unknown movements than the conventional methods, but the same results with known movements.

2.3 Subscriber Profile

The movement pattern function m(t) is a discrete function, it consists of samples $\{m(t_i)\}$. The time intervals t_i-t_{i-1} may be constant or non constant depending on the recording method. The first method is to register each location update generated from the mobile terminal when changing the location area. The location update messages arrive in non deterministic times. The second variant takes samples in constant time intervals by looking in the networks location table.

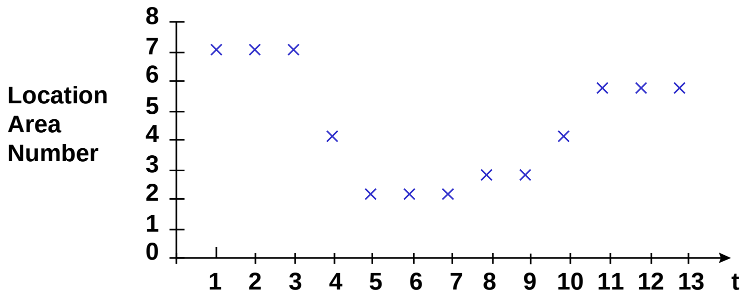


Figure 1: Subscriber Profile (Example)

In this examination we use only constant time intervals to build the user profile. A mobile network with 20 location areas was chosen, numbered from 1 to 20. We constructed a subscribers movement pattern (s. Figure 1)in this exemplary mobile network by writing down the locations regularly six times a day for four weeks, that means we obtain 168 location values. The sample rate may be higher depending on the subscribers movement and location area size. The absolute time values, e.g. days and weeks, don’t constitute a restriction of the generality. With these order of location area numbers, transformed into a set of input and output patterns, we carried out our investigations.

3 Input and Output Vectors

3.1 Contents

The contents of the input and output vector depends one the neural network type and on our different studies to find a relation between prediction output and contents of the vectors. Feedback neural networks have the ability to remember the order in which the input patterns are presented during the training, because of their backward connections. With these networks, it is only necessary to give the last known location area number l as input vector (1).

$$\vec{i} = (l); \quad l = 1, 2, ..., 20 \tag{1}$$

Feed-forward neural networks are memory-less, that means their output is independent of the previous inputs. For this neural network architecture a modified input vector is appropriate. One possibility is the presentation of the N last known locations (2). N has to be varied in our examination, to get the best prediction probabilities.

$$\vec{i} = (l_i, l_{i-1}, \dots, l_{i-N}); \quad l = 1, 2, \dots, 20 \tag{2}$$

In addition to the last known locations of the subscriber, other information can be given to the neural network. For example, we considered the absolute time, in form of a couple of day and hour (3). This information may be useful, because it may be correlated with the subscriber habits. The time information in the input vector is useful with both types of neural networks.

$$\vec{i} = (l_i, l_{i-1}, \dots, l_{i-N}, t); \quad l = 1, 2, \dots, 20 \tag{3}$$

The output-information is an other important topic. We want to obtain as output information only the location area number, in our example a value between 1 and L. The time information is not necessary in the output vector because we studied only a prediction for the next time interval t+Δt. If the prediction should be extended to the prediction of the location at the time t+kΔt the time information has to be included in the input vector. A second information which will be useful in the output vector are the probabilities that the subscriber is located in the predicted location areas.

3.2 Encoding of the input and output information

An input pattern is a set of activation values of the neurones belonging to the input layer of the neural network. The conversion of the real data into the input pattern is called encoding. There are several possibilities to present the input data to the neural network in a appropriate form. We studied the quasi-continuous and the binary coding.

Quasi-continuous representation means the real input data, e.g. location area number and time, are encoded with rational values in a certain interval, e.g. [0;1] in steps of (1/l). In this configuration the neural networks can be build with only one neurone in the input layer.

Binary encoding means the real input data are encoded with the values ‘0’ and ‘1’. In this case the amount of neurones in the input layer is equal to the amount of location areas plus the time values for one period. For this reason the total number of neurones in the input layer rises with the size of the modelled mobile network and the sample rate of the subscriber movement function in case of binary encoding.

The encoding of the output vector corresponds to the encoding of the input vector, with the difference that the time has not to be considered. Binary encoding of the output information means the predictions consists of one or more location area numbers without any probability information. Whereas quasi-continuous encoding is able to deliver probability information corresponding to the predicted location area number. This information can be used for multiple paging attempts. The disadvantage of this output layer structure is that each location area number needs to be coded with one output neurone.

Following the multiple paging method is described. The mobile system first pages the location area corresponding to the neurone with the greatest activation value. If the subscriber does not respond to the paging, the system tries the location areas corresponding to the next activation values in order of falling activation, until the subscriber is found or a maximal number of attempts is reached.

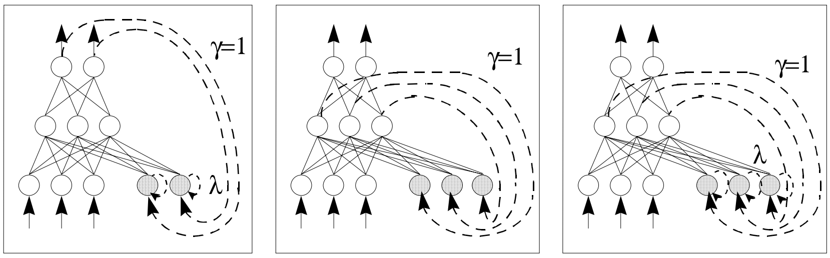
4 Examined Neural Network Architectures

Representing the feedback type we chose Jordan-, Elman- and Hierarchical Elman-Networks (s. Figure 2) and additionally we examined different feed-forward networks structures. The learning algorithms studied were the standard back-propagation algorithm, back propagation with momentum term, quickprop and resilient propagation.

The first step of the examination was the determination of the best specific network parameters, like learning, updating and initialising parameters. To avoid complicating the investigations, no input or site functions have been used, and neither activation nor output functions have been changed during the simulation. We chose the tangens hyperbolic function as activation and the identity function as output function.

The optimisation of the above mentioned parameters was carried out with one network architecture and was proved later with the different architectures. To evaluate the training results, we used the Sum of Squared Errors.

The second step was the simulation with the best parameters and different network architectures. The first simulations had been done with Elman networks, we obtained the following results. Binary coding of the input data showed the best generalisation results. The input vector with time information led to lower errors. The best learning algorithm was the resilient propagation. Variation of the amount of neurones in the hidden layer showed no significant changes in the error criterion.



Jordan Networks $\lambda \in [0,1]$ | Elman Networks | Hierarchical Elman Networks $\lambda \in [0,1]$

Figure 2: Examined Neural Network Architectures

The testing of Hierarchical Elman networks did not show better results than Elman networks, for this reason we did not incorporate them in the further investigations. The simulations supports the results from the Elman networks.

The next neural network structure studied was the Jordan network type. Rising amount of neurones in the hidden layer improves the results in this case. The other result, like learning with resilient propagation, binary coding and includingtime information, from Elman networks are confirmed. The simulation with the same variation of parameters showed no significant differences to the results of the Elman networks.

Feed-forward networks delivered the following results. The consideration of time in the input vector is very important to gain sufficient prediction results. Higher amount of neurones in the hidden layer improves the prediction. The best learning algorithm was back propagation with momentum term, the obtained results improved with rising of N, the last known locations in the input vector.

In our examination feed-forward networks delivered better results than the feedback networks with the chosen input vector and amount of training data. This is an unexpected result that suggests to use other input data and other learning algorithms in future. The feed-forward networks produce better results with more neurones in the hidden layer. The best learning rule back-propagation with momentum term showed better generalisation results with a rising number of historical states in the input vector.

At last we trained then the best neural network for 20000 learning epochs in order to test it with new profiles in the recall phase.

5 Prediction Results

5.1 Results with Neural Networks

After the training of the best feed-forward network, we tested it in the recall phase with changed profiles. The first profiles were a one day and the second a half-day shifting of the original subscriber profile on the time axis. An second studied case was a non moving mobile station in a previously visited location area. First results are promising; they showed correct prediction probabilities around 80% within three paging attempts.

With the first test profile (s. Figure 3, Profile 1) a one day shifting, the system was able to find the right location area after the first attempt with a probability of 77%. The probability after the third attempt reaches 93 %. The same trend was revealed by the results of the test with the half-day shifted profile (s. Figure 3, Profile 2). After the third attempt, the system was able to find the right location area with a probability of 85 %.

In the case of the constant profile in a previously visited location area (s. Figure 3, Profile 3), where the changes relatively to the original profile are no more insignificant, the results are still interesting (85 % probability after the 3rd attempt).

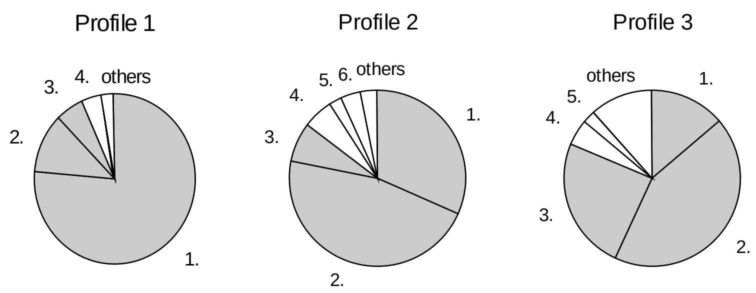


Figure 3: Results

A non moving mobile station in a location area never visited before leads to a prediction probability below 30% carrying out three paging attempts. We expect that this probability decreases further with rising number of location areas.

5.2 Comparison with Conventional Methods

The time shifting of the profile has no influence on the location prediction when using the state dependent method [2], because it’s independent of time. The moving patterns are known and for that reason the prediction is correct with 100%. Using the second conventional method with the time/location table [1] the results depend on the profile. If the new location has never been visited before at exactly that time the prediction will fail. If the location has been visited before the prediction probability lies below 50%, but the right location may be found in the second or third paging attempt.

The constant stay in an previously visited location area leads to bad results in the first attempts, because the state dependent method needs to store the new „movement“ pattern in the database and the time dependent method has no entry or an entry with low probability for this location.

The constant movement in an location area never visited before gives the same results for the state dependent method as in the case of constant stay in a previously visited area. The time dependent method fails in this case, because the probability of a never visited location area is zero, until the next update of the time/location table which contains the new location.

5.3 Discussion

The above results shows that the conventional methods are better in case of static movement patterns, like time shifting of a known profile. The neural network method has it’s advantages in case of dynamic motion patterns.

The variation of the movement pattern for which the neural networks deliver the best results, is a profile with light non-static variations witch differ from time to time. This is a characteristic we expect from real subscriber movements.

6 Conclusions

The first investigation shows promising results. They have to be studied more intensive to find the best networks structures. New advanced network types and learning algorithms are to be considered. Additional the influence of time in the input vector is a topic to optimise the prediction. Simulation with higher and lower weighted time influence on the prediction output has to be studied. For implementation purposes a combination of conventional and neural network methods will be investigated, to reduce the expenditure in real systems.

References

[1] S. Tabanne and R. Nevoux, “An intelligent location tracking method for personal and FPLMTS/UMTS communications”, ISS’95, 1995, pp.114-118

[2] G. Y. Liu and G. Q. Maguire Jr., “Efficient Mobility Management Support for Wireless Data”, IEEE VTC’95, 1995, pp. 902-906

[3] A. Zell, “Simulation Neuronaler Netze”, Reading, Addison-Wesley, 1994

Index:

mobility, location, prediction, subscriber profile, jordan networks, elman networks