Trajectory prediction in cellular networks using Discrete-time Markov chains

Mati Kärner Department of Computer Science
University of Tartu
Tartu, Estonia
mati.karner@ut.ee

Abstract—This paper provides one possible use-case how First-order Markov chains could be used to predict user trajectory in mobile network and explains the motivation behind such research. We developed a software prototype which consumes events from network operator provided data feed and produces a probability distribution characterizing possible paths a Mobile Equipment (ME) holder can take at some upcoming time interval. An anonymized real-world user data was used to evaluate the accuracy of the predictions. We give an overview of the results and implementational details, as well as provide possible solutions for the problems and difficulties experienced during the study.

I. INTRODUCTION

Mobile devices have become more widespread than ever. According to global data published by International Telecommunication Union 90 people out of 100¹ have mobile telephone subscriptions [1]. Innovation in technology and reduced production cost has made mobile phones with data connectivity and support for positioning systems, such as GPS, widely available. The fact, that people tend to carry cellphones with them wherever they go, gives rise to new exciting Location Based Services (LBS) and products. However, given the expensiveness of cellular data services and considering that free WiFi access is still rarity rather than commodity, metadata resulting from various GSM/UMTS/LTE network operations, such as voice calls, messages, radio operations triggered by ME etc, is still one of the cheapest retrievable data source for making meaningful conclusions about the characteristics of the network.

One particularly interesting data source, unfortunately unavailable for the period of study, is RSSI (Received Signal Strength Indication) feed. Among other parameters, this feed also includes the signal strength of the radio operation associated to some particular signal source. This in turn provides means to approximate the position of ME (using wireless triangulation). Numerous methods exist to model different aspects of RS (Radio Signal) propagation and user movement inside the network and this is a growing subject of research ([2][3][4][5] to name but a few). In this paper we only concentrate to Visitor Location Registry (VLR) feed that captures data from several types of events occurring in cellular networks, e.g. periodic location updates, CID (Cell ID) and LAC (Location Area Code) changes, UE (User Equipment) being switched on/off etc. We are specifically interested in those events that contain the CID, allowing us to model user movement as a series of state transitions between the cells.

Our aim was to experiment with Discrete-time Markov chains (DTMC) and gain hands on experience as well as deeper understanding of modeling stochastic processes. Having granted access to anonymized VLR feed of one of the mobile network operators in Northern-Europe, we developed an application that analyzes a specified interval from previously collected data of some cellphone user. This data is used to calculate necessary inputs for Markov chain predictor as well as picking out series of events (N-length random walk) on which we traverse from time ascending order and provide predictions for the next possible state transition. For sake of simplicity, we use one hour offsets as discrete time steps and choose 3 cells with the highest transition probabilities on each step. To evaluate the predictions we observe the next actual state on time step $t+1$ and inspect whether transitioning into this cell was predicted in time step $t$. A GUI tool for controlling the application and visualizing the path with corresponding predictions on the map was also developed.

This paper is organized as follows: Section II formulates the problem using First-order Markov chains. Section III covers the implementational part. Overview of the results is given in section IV. Section V summarizes the paper, highlights and provides possible solutions to some problems and difficulties experienced during the study.

II. PROBLEM FORMULATION

In this section we present how user mobility is formalized using First-order Markov chains. Each event in VLR feed in interpreted as state transition, and the CID of the event determines the state of the system at any given time. The state space is therefore the set of all CID-s.

Altogether, for
a) a set of $N$ states $S = \{s_1, s_2, ..., s_N\}$, where $s_i \in \mathbb{N}$,
b) state transition probability distribution matrix $A = \{a_{ij}\}$,
\[ a_{ij} = P(X_{t+1} = s_j \mid X_t = s_i) \quad 1 \leq i, j \leq N, \]
c) an initial distribution $\Pi = \{\pi(1), \pi(2), ..., \pi(N)\}$, where $\pi(i)$ indicates the probability of starting in state $s_i$,

estimate the probability distribution $P[X_{t+1}]$ at time $t+1$.  

¹119 out of 100 for developed countries
First-order Markov assumption states that only the knowledge of the present, not the past, is relevant for making predictions about the future states of the system. More formally

\[
P(X_{t+1} = s \mid X_t = s_t, X_{t-1} = s_{t-1}, \ldots, X_1 = s_1) = P(X_{t+1} = s \mid X_t = s_t).
\]

We’ll also assume that the Markov process under consideration is time homogeneous, that is, the processes that govern the system do not change over time \[6\]. This allows us to compute probability distribution for any subsequent time:

\[
P(X_{t+1} = j, X_t = j_t, \ldots, X_2 = j_1, X_1 = i) = (a_{j_{t-1}j_1}(a_{j_{t-2}j_{t-1}}(\ldots(a_{i,j_1}\pi(i)))))
\]

more generally

\[
P[X_{t+1:1} = \Pi A^t]
\]

**A. Initial distribution & state transition matrix**

Let \(\xi(i,j)\) be number of transitions from state \(i\) to \(j\) observed over some training set \(T_N\). One intuitive way how one could express the likelihoods incorporated into \(A\) is as follows:

\[
a_{ij} = \sum_k \frac{\xi(i,j)}{\xi(i,k)}
\]

Latter is simply a normalized count. Using similar approach for initial distribution yields

\[
\pi(i) = \sum_k \frac{\xi(i)}{\xi(k)}.
\]

### III. IMPLEMENTATION

Application prototype was built upon Node.js framework and consists of HTTP server and web application.

**A. Web application**

We used Open Layers library and their famous Stamen tiles for the map. Cell areas were drawn based on operator provided coverage area maps (multipolygons). Application also features a date range selector, that when submitted, queries HTTP Server for transition estimates together with ten last events, which are then visualized on the map.

**B. HTTP Server**

In addition to serving static assets, server also handles transition estimate requests, picks out the corresponding time interval from VLR log and passes them to DTMC predictor.

Majority of the functions in DTMC implementation operate on a data structure called state map, that maps the CID-s to their indexes in \(A\) as well as in initial distribution vector. State map also counts state transitions, which in turn is used to calculate initial distribution and state transition probability distribution matrix. We use Numberic library for Node.js for matrix-vector (or matrix-matrix) operations in (2).

![Fig. 1. Calculations with 50 days (7300 events) of VLR data. Sample 1.](image1)

![Fig. 2. Calculations with 50 days (3200 events) of VLR data. Sample 2.](image2)

![Fig. 3. Events from Figure 1. visualized on map. Polygons denote cell areas.](image3)

![Fig. 4. Events from Figure 1. visualized on map (zoomed). Polygons denote cell areas.](image4)
IV. RESULTS

As seen in Figures 1 and 2, the accuracy of the predictions grows together with the size of the training set. Both series seem to converge to certain value for training sets containing up to 45 - 50 days of data, i.e. \( \geq 2000 \) events. Data sets that contain events more localized to certain geographical area, without significant outliers, tend to give better predictions. This is the case between Sample 1 and Sample 2. Sample 1 contains a large cluster of events near Tartu (see Figure 3), but also has few smaller ones in Tallinn, near Narva and Hiiumaa, and multiple trails between Tartu and Tallinn. Sample 2, on the other hand, contains events originating from Harjumaa county and Tallinn. Assuming sufficiently large data sets, for non-localized data sets on average 40% of the events are among the ones predicted, 63% - 65% for localized sets.

Current implementation of the VLR feed caches around two months of events for each network subscriber. It would be interesting to run the application on a data set containing at least half a year of events and even more. These measurements, although possible, require better planning, since log has to be concatenated from multiple pieces, each captured after two month intervals.

V. CONCLUSIONS

In this paper we implemented an application predicting user movement between cellular network cells. The accuracy of the predictions was around 40% vs 65% for the unlocalized data sets and localized data sets respectively. For possible applications this is promising, but clearly not enough. Given the accuracy difference between localized and unlocalized training sets, it might be worth experimenting, if for given cell ID, the predictions made for the next time-step, only take into account the transition probabilities into the neighboring cells. This could be further improved by precalculating distributions over historical data of all all subscribers for each cell in the network. If the historical data for particular user doesn’t contain any transitions from specific cell to another, resulting zero probability either in initial distribution or state transition distribution matrix, the precalculated distribution would be a helpful clue. Another approach to improve the accuracy is to use higher order Markov chains, which build more “memory” into the states under observation, but are also more costly in terms of computation.

The future work for the current topic involves modelling the mobility of the user as series of state transitions, where each state also describes the Cartesian coordinates of the ME as well as the speed and velocity. A crucial mean for this is the signaling feed described in section I. This is, however, much more difficult task to tackle with, because of the various components included in mobility model, e.g. RSSI propagation. There are numerous interesting papers available for the given topic that all tend to highlight novel approaches and therefore further encourage writer to continue with the project.

REFERENCES