MobReduce: Reducing State Complexity of Mobility Traces

Fabian Hartmann, Christoph P. Mayer, Ingmar Baumgart
Institute of Telematics, Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
{hartmann,mayer,baumgart}@kit.edu

ABSTRACT

User traces are essential for analysis of human behavior and development of opportunistic networking protocols and applications. As user traces are collected with high granularity to apply them in diverse scenarios, they have a high complexity resulting from the large number of user states.

We present MobReduce: a methodology for reducing the number of states in user traces. We apply MobReduce to individually to GPS locations and WiFi sightings of the Nokia Mobile Data Challenge data set and show how to trade off state complexity vs. granularity.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Distributed networks, Network communications, Network topology, Store and forward networks, Wireless communication*

General Terms

Algorithms, Experimentation

Keywords

Trace, Mobility, State Reduction

1. INTRODUCTION

The rise of mobile devices has enabled a multitude of novel applications. Such applications exploit the mobility that devices are exposed to through their human owner to perform opportunistic communication when in mutual proximity [6]. They allow, e. g., to offload traffic from infrastructure-based networks with the help of opportunistic communication between mobile devices [8, 2, 13], or exploit physical contacts for social communication. The basis for the development of such applications is an understanding of the mobility and social characteristics exposed by the human owner upon the mobile device. Collection of mobility data has become essential for researching such characteristics [15], and extracting mobility and social patterns. In case of the mobility location patterns those have shown to follow a power-law behavior in inter-contact times[5, 10], as well as in the number of place visits [7, 12]. This indicates that a large number of places are visited very often. Reducing the data of such often visited places to one single state is the underlying idea of MobReduce.

Data is collected deliberately very general and fine-grained in experiments. This allows to apply the collected data in very different analysis. When analyzing the behavior through the collected trace it is beneficial to work with a reduced number of states to increase manageability. In this work we consider two cases of traces:

- Mobility traces are made up of GPS samples of multiple users collected over time. In this raw form, every GPS sample is considered one *state*.
- Mobility traces made up of WiFi sightings of multiple users collected over time. In this case, the definition of one *state* is more complex and detailed later.

Our methodology *MobReduce* transforms the set of states of multiple users into a common set of states for all users that is reduced in size. The goal of MobReduce is to transform a mobility trace into a manageable form that can be used as the basis for analysis, e. g. for analytical models like Markov chains.

This paper is structured as follows: Section 2 presents our methodology MobReduce for reducing state complexity of mobility traces. In Section 3 and Section 4 we apply MobReduce first to the GPS location and then to the WiFi trace collected within the Nokia Mobile Data Challenge [11]. We show how MobReduce can be applied to such traces and how to trade off the size of the state set vs. the granularity of the mobility behavior. Related work is presented in Section 5. Finally, Section 6 summarizes and gives an outlook on future work.

2. MOBREDUCE: MOBILITY STATE REDUCTION

We first describe a formalism for modeling user states and then present the MobReduce algorithm to perform state reduction. The algorithm is applied to both GPS traces and WiFi traces in Section 3 and Section 4, respectively. Symbols used in this work are listed in Table 1. Note, that we name a trace that contains several behavior like mobility, phone status etc. generally *user trace*. The part of the trace that contains location information and describes mobility behavior of users is named *mobility trace*.

We consider a *user trace* $M = \{U_1, U_2, \dots, U_n\}^d$ of n mobile users. U_i^d denotes the recorded user trace of one user i in *dimension* d of the trace. Dimensions make up different behavior of the user, e. g. location, phone status, messenger status, etc. The trace of one dimension k is de-

This material was prepared for the Mobile Data Challenge 2012 (by Nokia) Workshop; June 18-19, 2012; Newcastle, UK.

Symbol	Description
M	complete mobility trace
U_i^k	mobility trace of user i in dimension k
$s_{i,k}^{t_j}$	mobility state of user i in dimension k at time t_j
$S_{i,k}$	mobility state set of user i in dimension k
${S'}_k$	reduced, user-independent state set in dimension k
r_k	fuzziness of the reduced mobility states in dimension k
$\Delta_k(s, s')$	state distance function for dimension k
q	quality metric

Table 1: Symbols used in MobReduce.

fined as $U_i^k = \left(s_{i,k}^{t_1}, s_{i,k}^{t_2}, \ldots, s_{i,k}^{t_m}\right)$ and contains m_i *states* $s_{i,k} \in S_{i,k}$ of user i. The trace U_i^k defines the ordering of occurrences of states and their timestamps. Note, that U_i^k can contain the same state at different timestamps. Symbol $s_{i,k}^{t_j}$ determines the state of user i in dimension k at time t_j. Note that in this form of the original trace *all states are user-specific.* The *state set* of the user trace is $M = \{U_1^1 \cup U_1^2 \cup \ldots \cup U_1^d \cup U_2^1 \cup \ldots \cup U_n^d\}$. The size of M is the sum of all $m_i \cdot d$ states of all n users, denoted

$$|M| = \sum_{i=1}^{n} m_i \cdot d \tag{1}$$

The goal of MobReduce is to transform the trace $M = \{U_1, U_2, \ldots, U_n\}^d$ into a new trace

$$M' = \{(\{U'_1, U'_2, \ldots, U'_n\}, S')\}^d \tag{2}$$

so that

1. the overall number of states are reduced ($|M'| < |M|$), and
2. the states are common for all users, i. e. MobReduce performs a surjective mapping ($s_{i,d} \in S_i^d \to s'_d \in S'_d$).

This reduction step is performed per dimension. Reduction results in a reduced number of states ($|S'_d| < |S_{i,d}|$) and a reduced occurrence of time stamped state changes ($|{U'}_i^d| < |U_i^d|$). Every state has a corresponding dimension-specific fuzziness r_k that describes the accuracy of the state. Original states $s_{i,k}$ have a defined fuzziness of $r_k = 0$ and are considered the ground truth. States $s'_{i,k}$ resulting from the reduction step have a fuzziness $r_k > 0$.

The reduction algorithm is based on [1] and shown in Algorithm 1. A distance function $\Delta_k(s_{i,k}, s'_k)$ is defined that compares each $s_{i,k} \in M$ to the existing $s'_k \in S'_k$, using the user defined fuzziness r_k. If s lies within $s' \pm r_k$, it is transformed from s to s'. If s is not suitable for all existing states $s' \in M'$, a new state $s' := s$ is created. This is prone to over-fitting if the fuzziness r_k is chosen too restrictive. The distance function $\Delta_k(s, s')$ must be defined for each dimension separately.

2.1 Calculating the quality index

Reducing the number of states results in loss of accuracy. We have defined a quality metric to evaluate the effect of different parameters, i. e. the number of states and the configuration-specific fuzziness, on the accuracy. We define $Z_d^i = \{s \in S_{i,d} | \exists s' \in S'_d : \Delta_d(s, s') < r_d\}$ as all the original states from user i and dimension d that are transferable to the reduced user-independent state set S'_d. This version of the quality index represents the ratio of the number of original states that get represented in a reduced state, compared to the overall number of original states. The quality index for user i and dimension d is then:

Algorithm 1 MobReduce algorithm

```
DimensionsToReduce := $\{d_1, d_2, \ldots\}$
FuzzinessPerDimension := $\{r_{d_1}, r_{d_2}, \}$
$S' := \emptyset$
for all $U_i \in M$ do
    for all $U_i^d \in U_i$ do
        if $d \notin$ DimensionsToReduce then
            continue
        end if
        for all $s_{i,d}^{tj} \in U_i^d$ do
            bestDistance := $r_d$
            for all $s'_d \in S'_d$ do
                if $\Delta_d(s_{i,d}, s') <$ bestDistance then
                    $s'_{opt} = s'$
                    bestDistance := $\Delta_d(s_{i,d}^{tj}, s')$
                end if
            end for
            if bestDistance = $r_d$ then
                ▷ No suitable state found, create new $s'$
                ${U'}_i^d := {U'}_i^d \cup s_{i,d}^{tj}$
                $S'_d := S'_d \cup s_{i,d}^j$
            else
                ▷ Use $s'$ with least distance
                ${U'}_i^d := {U'}_i^d \cup {s'}_{opt}^{tj}$
            end if
        end for
    end for
end for
```

$$q_{i,d} = \frac{|Z_d^i|}{|U_i^d|} \tag{3}$$

Since the fuzziness r_d is not represented in $q_{i,d}$, we define an extended version of the quality index, using $\bar{\Delta}_{i,d}$ as the average value for the distances between all $s_{i,d}$ and their best matching s'_d. It holds $0 < \bar{\Delta}_d < r_d$. The closer $\bar{\Delta}_d$ is to r_d, the worse the overall quality should be valued, due to the increased average fuzziness. Hence we define the penalty factor $p = 1 - \frac{\bar{\Delta}_d}{r_d}$.

The improved quality index for user i and dimension d is:

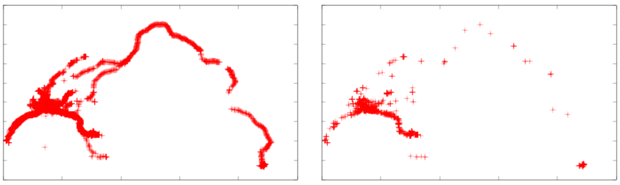
$$q_{i,d}^{*} = q_{i,d} \cdot p = \frac{|Z_d^i|}{|U_i^d|} \cdot \left(1 - \frac{\bar{\Delta}_d}{r_d}\right) \tag{4}$$

The overall quality index across all users for dimension d is the arithmetic mean across all users:

$$q_d = \frac{1}{n} \cdot \sum_i q_{i,d}$$

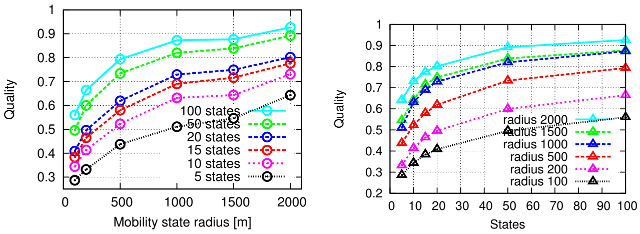
and respectively for q^*:

$$q_d^* = \frac{1}{n} \cdot \sum_i q_{i,d}^*$$



(a) Original states (b) Reduced states

Figure 1: Playground with exemplary reduction of the location states of one user. Every point represents one location state, irrespective of time.



(a) Mapping per state (b) Mapping per radius

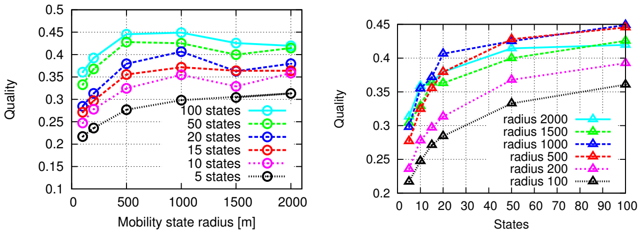
Figure 2: Quality of location state reduction, using q_{gps}.

3. APPLYING MOBREDUCE TO GPS LOCATION TRACES

We apply the methodology of MobReduce to the Nokia Challenge Data user trace [11] for the dimension of *location*. Within the dimension of location, a state $s^{t_j}_{i,\text{loc}} = (x, y)$ describes the position of user i at time t_j on a playground[1]. Fuzziness r_{loc} describes an allowable radius of the original location state to the new state when performing the reduction step. The distance function is defined as euclidean distance between the location states as $\Delta_{\text{loc}}(s, s') = \sqrt{(x_s - x_{s'})^2 + (y_s - y_{s'})^2}$

In case of location states, Algorithm 1 clusters sets of nearby location samples to a single location and only regards the arrival and leaving timestamps. This means that commonly visited locations over all users are extracted. Figure 1 shows an exemplary location trace reduction: Figure 1a shows the original location trace of one user. After applying MobReduce to the traces of all users, the resulting common state set is shown in Figure 1b.

We performed reduction of the Nokia Challenge Data location trace and analyzed the resulting quality metrics introduced in Section 2. Figure 2 shows the quality metrics for different numbers of states and fuzziness. For quality index q_d, we see that raising both the radius as well as the number of locations increases the overall quality. The growth in quality saturates over both the number of states and radius. Figure 3 shows the same evaluation using the quality metric q^*_d. It can be seen that an increase in the radius size does not provide continuous quality increase. Rather, at a radius size of around 1000 m, only an increase in the number of states provides further quality improvements. This suggests that an optimal setting exists which depends on the mobility behavior.



(a) Mapping per state (b) Mapping per radius

Figure 3: Quality of location state reduction, using q^*_{gps}.

In case of the Nokia Challenge Data set our evaluation indicates that for a radius of 1000 m and only 100 common states the quality is high enough to reflect the general mobility patterns of the user, while reducing the complexity of the trace heavily.

4. APPLYING MOBREDUCE TO WIFI SIGHTINGS

In the following we show how MobReduce can be applied to traces of WiFi sightings, which in turn can be used to model user locations[2].

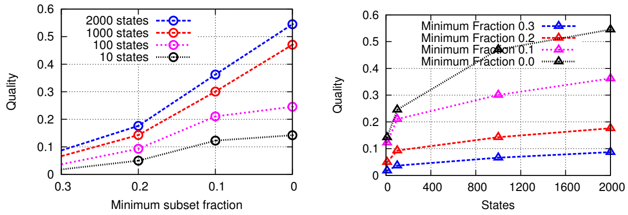
It is not our goal to emulate GPS tracking—like e. g. performed in [16, 14]—therefore we do not take WiFi signal strength into account. Using GPS coordinates results in an explicit state per timestamp. In contrast, several WiFi sightings might be possible at one timestamp. Therefore, it is not possible to define one WiFi sighting as one state. To model states we cluster WiFi sightings at one timestamp into one state. Furthermore, we map subsets of WiFi sightings to the original state which is made up of a superset of corresponding WiFi sightings.

In order to calculate the quality index for WiFi sighting traces q^*_{wifi}, we again require the maximum allowed fuzziness r_{wifi}, and an average fuzziness $\bar{\Delta}_{wifi}$ of reduced states. The task of defining fuzziness in the context of WiFi sightings is particularly interesting, since it differs from the straightforward radius idea that can be applied to GPS coordinates. As we do not take signal strength into account, a WiFi sighting is a strict binary decision. Note that this is in contrast to the continuous geographic distance used in GPS traces.

Each state represents a WiFi cluster, which is a unique combination of visible WiFis at the same time. We regard the WiFi clusters from S'_{wifi} as supersets, while normal sightings $s \in S_{i,wifi}$ are regarded as subsets. Fuzziness is defined based on the relative sizes of such subsets. Our definition is based on the assumption that the more WiFis of a cluster are visible, the closer the user is to the sweet spot which defines the state. For example, given a WiFi cluster of $\{A, B, C\} \in S'_{wifi}$ and a sighting of B and C at a given timestamp (i. e. 2/3 of the state's cluster), the user is closer to the state than at a time where only WiFi B is visible (i. e.

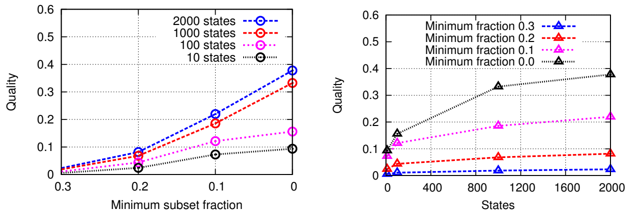
[1] For actual transformation we calculate the distance between the GPS coordinates using the Harvesine formula and define a 2D playground using the left/right/top/bottom-most locations.

[2] This method is for example used by Google and Apple to detect a user's current location without GPS tracking, see [4] for a comprehensive primer on this subject.



(a) Mapping per state (b) Mapping per fraction

Figure 4: Quality of WiFi state reduction, using q_{wifi}.



(a) Mapping per state (b) Mapping per fraction

Figure 5: Quality of WiFi state reduction, using q^*_{wifi}.

1/3 of the state's cluster). Therefore, r_{wifi} is defined via a minimum fraction $f \in [0..1]$ that must be exceeded, where f is defined as the ratio of visible WiFis during a sighting s to the WiFi clusters given by S'_{wifi}. Every $s \in S_{i,wifi}$ is thus mapped to the $s' \in S'_{wifi}$ which yields the highest f. If there exists no s' where the minimum value for f is exceeded, s is considered to be outside of the maximum allowed fuzziness.

We use a minimum subset fraction as quality metric, i. e. we *require* that a minimum fraction of the WiFi sighting subset is fulfilled. This corresponds to the maximum allowed radius definition used in Section 3. Using this minimum subset fraction and the number of states, we evaluate the quality metrics q_{wifi} and q^*_{wifi} shown in Figure 4 and Figure 5, respectively. The subset fraction employs a ">" relation, i. e. a minimum subset fraction of 0.0 requires at least one WiFi sighting of the original cluster. Note, that a high minimum fraction value is more restrictive, in contrast to a high radius value that is less restrictive. Therefore, the x-axis in Figure 4a and Figure 5a are inverted. Both Figure 4 and Figure 5 show the same characteristics. However, our quality metric q^*_{wifi} is more restrictive.

Using a high number of states results in linear increase under a less restrictive subset fraction. Indicating that a rather high number of states is required for reducing WiFi traces. The less restrictive the minimum subset fraction is chosen, the higher is the benefit from a larger number of states.

Note, that the quality metrics between GPS trace and WiFi sighting trace can not be easily compared, as the respective fuzziness definitions are not comparable.

5. RELATED WORK

Ashbrook and Starner presented in [1] an approach for reducing fine-grained GPS data to a smaller number of locations. Those locations are used to perform predictions of future movements. In comparison, MobReduce does not solely focus on GPS data but provides a more general methodology for state reduction in multi dimensional user traces.

In [9] Hummel and Hess present an approach for GPS-based movement prediction. While they do not focus on state reduction, they augment their data with additional semantic information like "home", "shopping", or "evening location" in order to improve the prediction model. In its current form MobReduce does not integrate semantic information. As such, the work of Hummel and Hess could be used to extend the MobReduce approach.

6. CONCLUSION AND OUTLOOK

To cope with the large amount of data in user traces we presented *MobReduce*. Using clustering MobReduce aggregates and reduces the states to a manageable set, based on a small number of parameters. We have shown that MobReduce provides a general methodology that we exemplary applied to GPS traces and traces of WiFi sightings. For both traces we defined quite different fuzziness and quality metrics to integrate them with MobReduce. We analyzed the complexity vs. granularity trade off in several studies using those traces. For GPS traces we have shown that there exists an optimal radius over which the quality can only be optimized using a larger number of states.

In future work we will compare different clustering algorithms and analyze the behavior of MobReduce for further dimensions like phone status or messenger status. Our final goal is to build multi-dimensional user models with low complexity. We will use those models to apply them for the development of opportunistic social networking systems. This also includes deriving possible social relationships from multi-dimensional states. For example, if two users are often in the same location and they appear in each others call logs, this might be an interesting information in regards of routing and data storage. Our SODESSON project [3] focuses on leveraging these kinds of information.

Acknowledgment

This work is supported by the "Concept of the Future" of KIT within the framework of the German Excellence Initiative in the SODESSON project, and by the Landesstiftung Baden Württemberg gGmbH within the SpoVNet project.

7. REFERENCES

[1] D. Ashbrook and T. Starner. Using GPS to Learn Significant Locations and Predict Movement Across Multiple Users. *Personal and Ubiquitous Computing*, 7(5):275–286, Oct. 2003.

[2] X. Bao, U. Lee, I. Rimac, and R. R. Choudhury. DataSpotting: Offloading Cellular Traffic via Managed Device-to-device Data Transfer at Data Spots. *ACM SIGMOBILE Mobile Computing and Communications Review*, 14(3):37–39, Dec. 2010.

[3] I. Baumgart and F. Hartmann. Towards Secure User-centric Networking: Service-oriented and Decentralized Social Networks. In *Proceedings of IEEE Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW)*, pages 3–8, Ann Arbor, MI, USA, Oct. 2011.

[4] A. Cavoukian and K. Cameron. Wi-Fi Positioning Systems: Beware of Unintended Consequences. *Information and Privacy Commissioner Discussion Papers*, June 2011.

[5] A. Chaintreau, P. Hui, C. Diot, R. Gass, J. Scott, and J. Crowcroft. Impact of Human Mobility on Opportunistic Forwarding Algorithms. *IEEE Transactions on Mobile Computing*, 6(6):606–620, June 2007.

[6] M. Conti and M. Kumar. Opportunities in Opportunistic Computing. *IEEE Computer*, 43(1):42–50, Jan. 2010.

[7] M. C. González, C. A. Hidalgo, and A.-L. Barabási. Understanding Individual Human Mobility Patterns. *Nature*, 453(7196):779–782, June 2008.

[8] B. Han, P. Hui, M. V. Marathe, G. Pei, A. Srinivasan, and A. Vullikanti. Cellular Traffic Offloading Through Opportunistic Communications: A Case Study. In *Proceedings of International Workshop on Challenged Networks (CHANTS)*, pages 31–38, Chicago, IL, USA, Sept. 2010.

[9] K. A. Hummel and A. Hess. Movement Activity Estimation and Forwarding Effects for Opportunistic Networking Based on Urban Mobility Traces. *Wireless Communications and Mobile Computing*, Mar. 2012. digital.

[10] T. Karagiannis, J.-Y. L. Boudec, and M. Vojnović. Power Law and Exponential Decay of Inter Contact Times between Mobile Devices. In *Proceedings of International Conference on Mobile Computing and Networking (MobiCom)*, pages 183–194, Montreal, QC, Canada, Sept. 2007.

[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 *Proceedings of Mobile Data Challenge Workshop (in conjunction with International Conference on Pervasive Computing)*, Newcastle, UK, June 2012.

[12] I. Rhee, M. Shin, S. Hong, K. Lee, and S. Chong. Human Mobility Patterns and Their Impact on Routing in Human-Driven Mobile Networks. In *Proceedings of ACM Hot Topics in Networks (HotNets)*, Atlanta, GA, USA, Nov. 2007. digital.

[13] J. Whitbeck, Y. Lopez, J. Leguay, V. Conan, and M. D. de Amorim. Relieving the Wireless Infrastructure: When Opportunistic Networks Meet Guaranteed Delays. In *Proceedings of IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM)*, pages 1–10, Lucca, Italy, June 2011.

[14] Z. Xiang, S. Song, J. Chen., H. Wang, J. Huang, and X. Gao. A Wireless LAN-based Indoor Positioning Technology. *IBM Journal of Research and Development*, 48(5.6):617–626, Sept. 2004.

[15] E. Yoneki. The Importance of Data Collection for Modelling Contact Networks. In *Proceedings of International Workshop on Social Computing with Mobile Phones and Sensors: Modeling, Sensing and Sharing (SCMPS)*, pages 940–943, Vancouver, Canada, Aug. 2009.

[16] M. Youssef, A. Agrawala, and A. U. Shankar. WLAN Location Determination via Clustering and Probability Distributions. In *Proceedings of IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 143–151, Dallas-Fort Worth, TX, USA, Mar. 2003.