Collaborative Mobile Gossip Learning

Árpád Berta

Supervisor

Dr. Márk Jelasity

Department of Computer Algorithms and Artificial Intelligence

Doctoral School of Computer Science

University of Szeged

Summary of PhD Thesis

Szeged

2020

Introduction

Over the past few decades, we have witnessed an explosive growth of mobile and smart devices and their widespread use. These devices are present in almost every aspect of our daily lives. This trend has led to numerous intelligent applications based on data mining. It is usually performed over collected data at a central location. This conventional process has become more and more problematic due to the increasing public awareness of the privacy issue. In the last few years, new privacy protection laws have come into force. For this reason, there is an increasing interest in methods that allow us to keep our private data in our devices and process them using collaborative algorithms.

There are, of course, many ways to address this challenge. We opted for gossip learning [8] due to the fact that it is fully decentralized, hence no central server is needed. Nodes exchange and aggregate models directly. This makes scalability significantly cheaper than the alternatives. It is a good opportunity for startups or communities with low budgets to provide robust intelligent smartphone services. It can serve the common good (e.g. public healthcare and public education). Although we focus on collaborative mobile platforms, gossip learning applications can be found in smart metering and over Internet of Things platforms as well.

In gossip learning, privacy is not guaranteed by default, but it is much easier to achieve. Moreover, theoretical notions of privacy such as differential privacy can be included as well. From this point of view, we propose a random walk service that can maintain a single walk and drive down the privacy budget. Our long-term goal is to provide a fully open collaborative environment where those who provide data can enjoy the benefits of mining the collective data of the community. To realize this goal, we propose a multiple random walk service for maintaining independent decentralized tasks that might belong to different users. We also introduce a smartphone trace based on collected data in order to obtain more realistic simulations. It contains network properties and patterns of user behavior. We can also take into account the improving performance of the gossip learning. Hence, it is crucial to minimize the communication cost by reducing the dimensionality of the data mining tasks. For this reason, we propose a number of robust and efficient approaches to handle this.

Table 1: The relationship between theses and the corresponding publications (where • and ∘ indicate the core and the related publications, respectively).

	Thesis 1	Thesis 2	Thesis 3
P2P 2014 [3]	•	∘	∘
TIST 2016 [7]	•	∘	∘
ICCGI 2017 [9]	•		
SCN 2018 [5]	•		
DAIS 2019 [11]	•		
PDP 2016 [1]	∘	•	
JOWUA 2016 [6]	∘		•
PDP 2017 [2]	∘		•
ESANN 2014 [4]		∘	
IJASO 2018 [10]	∘		

Summary of the Thesis Results

The main aim of this study was to tackle a number of diverse problems on mobile gossip learning in order to make gossip learning more suitable for performing distributed data mining. Here, we give a brief summary of each thesis. We mainly focus on results that the author regards as his main contributions. Table 1 lists the relationship among the relevant publications and the theses. Moreover, we mark the difference between core and related publications. In Thesis 1, we introduced a smartphone trace that is also presented in chapters 3 and 6 of the dissertation. The results of our dimension reduction in Thesis 2 correspond to Chapter 4 of the dissertation. The random walk management related results in Thesis 3 are detailed in Chapter 5 as well.

Thesis 1: Smartphone Trace

We proposed a real trace of smartphone user behavior for simulating node churn in distributed smartphone network. The presented trace was collected by a locally developed Android app that provides the user with information about the current network environment of the phone. We had traces of varying lengths harvested from thousands of different users. We divided these traces into

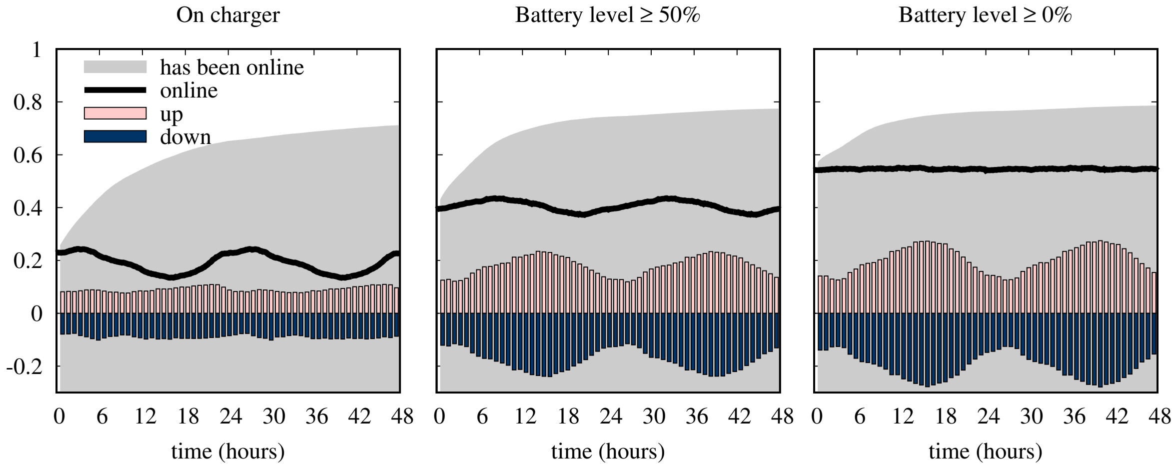

Figure 1: Proportion of users online, and proportion of users that have been online, as a function of time. The bars denote the proportion of the simulated users that log in and log out (shown as a negative proportion), respectively, in the given period.

one-day (or 2-day with a one-day overlap) segments that resulted in nearly 40,000 segments in total. With the help of these segments, we are able to simulate a virtual period of up to one day (or two days) by assigning a different, randomly selected segment to each simulated node. When the pool of segments runs out we re-initialize the pool with the original segments and continue the sampling without replacement. The proposed trace-based simulations provide a more realistic evaluation for fully distributed protocols.

The requirement of being on a charger or the expected minimum level of battery power may be a free parameter of a trace-based evaluation. The observed churn pattern is shown in Figure 1 based on 2-day periods we identified. We can implement different scenario by defining requirements of availability in the network. To ensure an algorithm is phone and user friendly, we can define a user to be available when the user has a network connection and the phone is connected to a charger. From the point of view of churn, though, the worst case is when phones with any battery level are allowed to join, because this results in a more dynamic scenario.

For the last couple of years, we have evaluated most of our proposed gossip protocols by simulating them on top of this real smartphone trace. We did likewise with our dimension reduction protocols and random walk management protocols We presented a fully distributed mini-batch gradient descent method that creates a temporary group of random nodes that form the mini-batch.

We did this by building a rooted and trunked overlay binomial tree. Here, it was crucial to demonstrate that the proposed overlay tree building is practically viable. Hence in our experiments, we simulated this on top of the smartphone churn trace. As a result it was clear that only a very small proportion of tree building attempts was unsuccessful.

In addition, we proposed a time-inhomogeneous Markovian model as an alternative to simulating churn. It is based on the collected data in which the conditional probability distributions of session lengths are captured by a set of actual observations in the data that we resample when creating synthetic traces of users to model churn. We validated this model in multiple ways. We found that the model captures observed availability as well as the behavior of push-pull gossip broadcast.

Since the initial release of our data collection app we have collected a very large trace involving millions of individual measurements. This amount of data naturally contains noise and some incorrect records. We described lessons learned on data collection over the years. In this context, we took great care to clean this data. We proposed a method to correct failed NAT measurements. And we extended the application to collect data related to direct peer-to-peer (P2P) capabilities based on a basic WebRTC implementation. Now, our trace contains locally observable attributes such as battery status and network availability, STUN measurements, along with direct P2P connection data. In this unique combination, we can combine these sources of data to be able to predict things like P2P connection success, or to simulate distributed protocols over overlay networks of smartphones. We presented a brief introduction on its base statistics and properties.

Main contributions of the author are:

- A trace for simulating realistic smartphone churn.
- An introduction to base statistics and properties of a smartphone trace.
- A time-inhomogeneous Markovian model as an alternative for simulating churn.
- An empirical evaluation of a fully distributed mini-batch gradient descent that is based on a secure sum protocol and construction of a k-long-trunked binomial overlay tree on top of the smartphone trace.

- A data cleansing method to correct failed NAT measurements.

- The corresponding paper are [3, 5, 7, 9, 11]:

  Árpád Berta, Vilmos Bilicki, and Márk Jelasity. Defining and understanding smartphone churn over the internet: a measurement study. In *14-th IEEE International Conference on Peer-to-Peer Computing (P2P)*, pages 1–5, Sept 2014

  István Hegedűs, Árpád Berta, Levente Kocsis, András A. Benczúr, and Márk Jelasity. Robust decentralized low-rank matrix decomposition. *ACM Transactions on Intelligent Systems and Technology*, 7(4):62:1–62:24, May 2016

  Zoltán Szabó, Vilmos Bilicki, Árpád Berta, and Zoltán Richárd Jánki. Smartphone-based data collection with Stunner using crowdsourcing: Lessons learnt while cleaning the data. In *The Twelfth International Multi-Conference on Computing in the Global Information Technology (ICCGI 2017)*, pages 28–35, Jul 2017

  Gábor Danner, Árpád Berta, István Hegedűs, and Márk Jelasity. Robust fully distributed mini-batch gradient descent with privacy preservation. *Security and Communication Networks*, 2018:15, 2018

  Zoltán Szabó, Krisztián Téglás, Árpád Berta, Márk Jelasity, and Vilmos Bilicki. Stunner: A smart phone trace for developing decentralized edge systems. In José Pereira and Laura Ricci, editors, *Proceedings of the 19th International Conference on Distributed Applications and Interoperable Systems (DAIS 2019)*, pages 108–115, Cham, 2019. Springer International Publishing

Thesis 2: Dimension Reduction Methods

Here, we proposed a number of robust and efficient decentralized approaches for dimension reduction in a system model like this, where each network node holds only one data record. Decentralized learning algorithms are very sensitive to the size of the raw data records due to the resulting large communication cost. This can, in the worst case, even make decentralized learning infeasible. Dimension reduction is a key technique used to compress data and to obtain small models.

The first presented algorithm builds on searching for good random projections. It is very cheap to generate a random matrix and it can be communicated by sending just the corresponding random seed. Because of this, we had the possibility of evaluating a lot of different random matrices and searching for the best one for a fixed machine learning problem (data and a learning algorithm). As

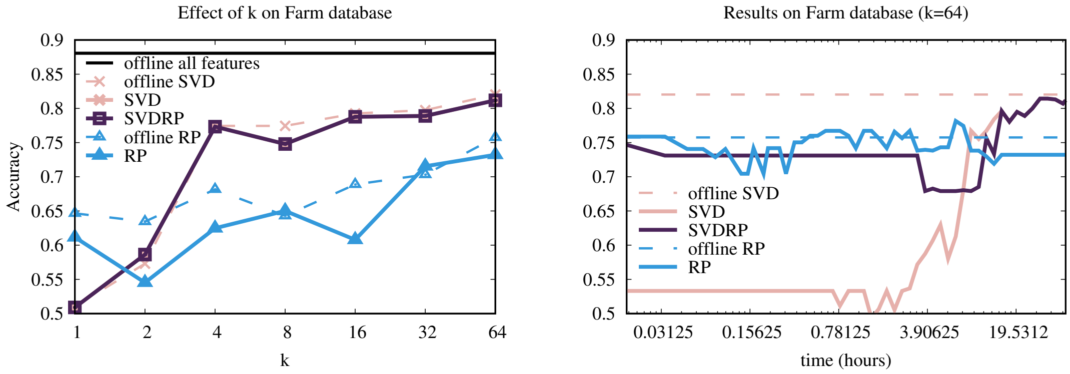

Figure 2: Comparison of dimension reduction algorithms accuracy after two days of simulated time as a function of k (left-hand side) and accuracy as it evolves in time with a fixed $k = 64$ (right-hand side).

for the size of a single message in the Random Projection protocol, the nodes send two models in a message that contains only the seed of the random projection matrix, and the model parameters, which are of size $O(k)$ in the case of linear learning algorithms. Because k, the free parameter of dimension reduction, is typically a small number, this leads to a practically negligible message size. In contrast to this, in the distributed singular value decomposition (SVD, a state-of-the-art algorithm for dimension reduction protocol) the whole projection matrix of size $O(k \cdot d)$ needs to be sent in each message. This can be very large since d is potentially in the order of millions.

We made a detailed experimental comparison of the proposed algorithms and compared them with SVD (see Figure 2). We based our experiments on the above-described trace of real mobile phone usage. We concluded that our method based on selecting good random projections is preferable and provides good quality results when the output is required on a very short timescale, within tens of minutes. However, the SVD algorithm converges in a time proportional to the original dimensionality of the problem, which can be quite slow. The dimension reduction quality of SVD is somewhat higher than that of our method.

We also presented a hybrid method that combines the advantages of random projections and SVD. We ran the two algorithms in parallel, and we did not assign the same bandwidth quota to each. Instead, we allowed the SVD algorithm to consume almost all the bandwidth quota, and assigned only 1% of it to random projection selection. The rows of SVD projection matrix converged in a sequential order, and in the converged state the rows were pairwise orthogonal. Our basic idea was that we define a new projection matrix that will use the converged rows from SVD matrix and will fill in the rest of the rows from random projection matrix. We demonstrated that this hybrid method provides a good performance over all timescales.

Main contributions of the author are:

- An algorithm that builds on searching for good random projections.
- A hybrid method that combines the advantages of random projections and SVD.
- The state-of-the-art distributed SVD on dimension reduction was evaluated in a comparative study.
- The corresponding paper is [1]:
  Árpád Berta, István Hegedűs, and Márk Jelasity. Dimension reduction methods for collaborative mobile gossip learning. In *2016 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)*, pages 393–397, Feb 2016

Thesis 3: Management of Random Walks

Here, we introduced two random walk management services for very different problems. Both of our presented studies are based on gossip learning in the sense that we focus on stochastic gradient descent (SGD) algorithms that are implemented via a random walk of the evolving model over the network. Also, we have three requirements for such a random walk management service: agility, longevity and efficiency. We simulated both service over the above-described smartphone trace.

Our first contribution is the Single Random Walk Service. It relies on maintaining a very small shared state through gossip that describes the progress of the random walk. Based on this shared state, each node decides independently whether to restart the random walk. Nodes that store a more recent state restart the walk with a higher probability. The approach is fully decentralized and only incurs a relatively small overhead if the random walk has a large state. We propose to implement SGD on top of this fault-tolerant single random walk service to keep the privacy budget low. Hence in this case, every data record is visited only a limited number of times. It offers the choice of differential privacy by adding the appropriate noise level.

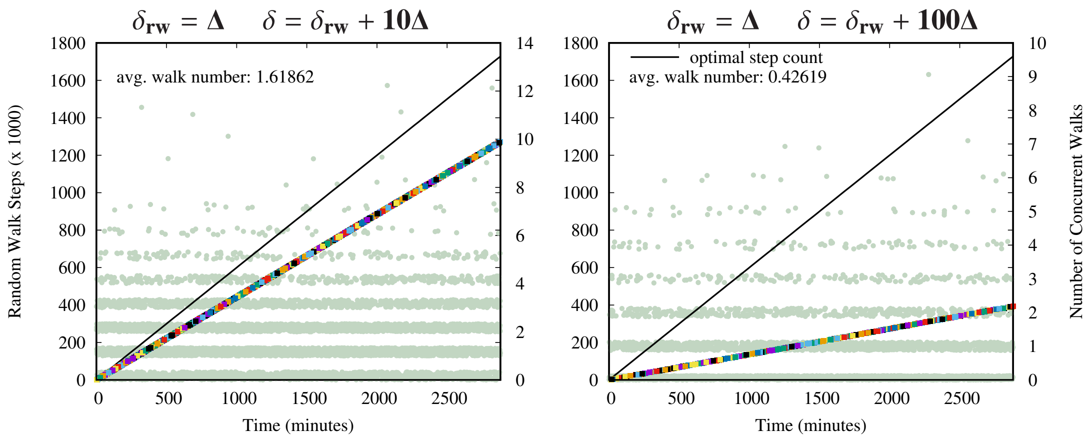

Figure 3: Experiments on Single Random Walk Service with 5% drop probability. The number of random walks is shown as dots (integers, translated slightly vertically by random noise to illustrate density) and the step count of the oldest random walk is represented as square, different shades indicating different random walks.

We demonstrated that the proposed method meets our requirements in a network of mobile smartphones. We found that the protocol is robust to its main parameter, the timeout threshold δ, which determines when a random walk is considered dead. As we see in Figure 3, we got an acceptable performance even in an unrealistic extreme scenario where we artificially removed random walks with a 5% probability in each step. In this case, this protocol is more sensitive to δ, but with a sensible setting a good compromise can be achieved between efficiency and agility.

Our second contribution is the Multiple Random Walk Service. Here we focus on the problem of maintaining $O(n)$ random walks over an overlay network, where the random walks represent independent decentralized tasks that might belong to different users. Essentially, our solution is made up of three conceptual levels. At the first (lowest) level a local mechanism is implemented. Here, after completing a random walk hop, every node monitors the success of the next hop. The problem gets escalated to level two when the monitoring node fails before it can detect the failure of the walk. The node performing the current hop therefore monitors the monitoring node (called the supervisor) and invites a new supervisor if the current supervisor fails. In the case of a level two failure, the supervisor broadcasts the restarting request that will eventually reach those nodes that have fresh versions of the payload. After a successful restart another broadcast is sent about the success, which prevents any new attempts at restarting. The third (and final) level is implemented by the central control carried out by the owner of the walk.

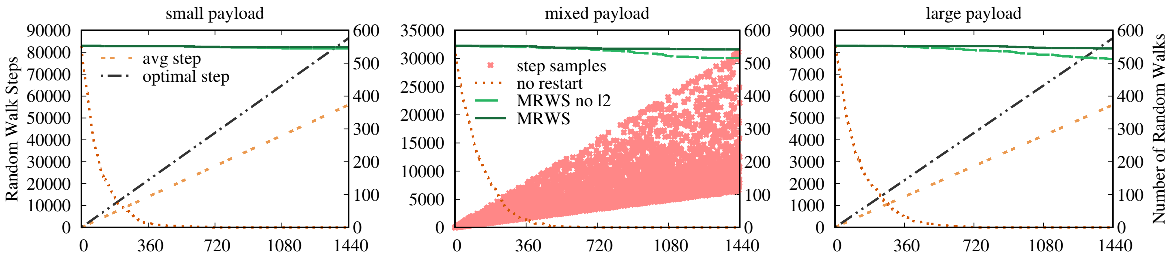

Figure 4: Experimental on the results of Multiple Random Walk Service with small, mixed and large payload, with the same amount of walks as that of the average online nodes

We demonstrated that in all the scenarios we tested, the vast majority of failures are handled at the lowest level, which is purely local and therefore scalable. Only a small fraction of the problems get escalated to level two, which is based on a broadcast primitive. In this case the cost of broadcast messages was shown to be almost negligible due to the small number of failure cases that reach this level. Thus, the overhead introduced by level two is small relative to the communication cost of the random walks. Level three, the central control by the task owner, was reached only a few times during all our simulations. We also demonstrated that the random walks are kept alive and the speed of the random walks is close to optimal. These results are shown in Figure 4.

Main contributions of the author are:

- A method for managing a single random walk.
- A method for managing multiple random walks.
- The corresponding paper are [2, 6]:

  István Hegedűs, Árpád Berta, and Márk Jelasity. Robust decentralized differentially private stochastic gradient descent. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, 7(2):20–40, June 2016

  Árpád Berta and Márk Jelasity. Decentralized management of random walks over a mobile phone network. In *2017 25th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)*, pages 100–107, Mar 2017

References

[1] Árpád Berta, István Hegedűs, and Márk Jelasity. Dimension reduction methods for collaborative mobile gossip learning. In *2016 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)*, pages 393–397, Feb 2016.

[2] Árpád Berta and Márk Jelasity. Decentralized management of random walks over a mobile phone network. In *2017 25th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)*, pages 100–107, Mar 2017.

[3] Árpád Berta, Vilmos Bilicki, and Márk Jelasity. Defining and understanding smartphone churn over the internet: a measurement study. In *14-th IEEE International Conference on Peer-to-Peer Computing (P2P)*, pages 1–5, Sept 2014.

[4] Árpád Berta, István Hegedüs, and Róbert Ormándi. Lightning fast asynchronous distributed k-means clustering. In *22th European Symposium on Artificial Neural Networks*, ESANN 2014, pages 99–104, 2014.

[5] Gábor Danner, Árpád Berta, István Hegedűs, and Márk Jelasity. Robust fully distributed mini-batch gradient descent with privacy preservation. *Security and Communication Networks*, 2018:15, 2018.

[6] István Hegedűs, Árpád Berta, and Márk Jelasity. Robust decentralized differentially private stochastic gradient descent. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, 7(2):20–40, June 2016.

[7] István Hegedűs, Árpád Berta, Levente Kocsis, András A. Benczúr, and Márk Jelasity. Robust decentralized low-rank matrix decomposition. *ACM Transactions on Intelligent Systems and Technology*, 7(4):62:1–62:24, May 2016.

[8] Róbert Ormándi, István Hegedűs, and Márk Jelasity. Gossip learning with linear models on fully distributed data. *Concurrency and Computation: Practice and Experience*, 25(4):556–571, 2013.

[9] Zoltán Szabó, Vilmos Bilicki, Árpád Berta, and Zoltán Richárd Jánki. Smartphone-based data collection with Stunner using crowdsourcing: Lessons learnt while cleaning the data. In *The Twelfth International Multi-Conference on Computing in the Global Information Technology (ICCGI 2017)*, pages 28–35, Jul 2017.

[10] Zoltán Szabó, Vilmos Bilicki, Árpád Berta, and Zoltán Richárd Jánki. Smartphone-based data collection with Stunner, the reality of peer-to-peer connectivity and web real-time communications using crowdsourcing: Lessons learnt while cleaning the data. *International Journal On Advances in Software*, 11(1-2):120–130, 2018.

[11] Zoltán Szabó, Krisztián Téglás, Árpád Berta, Márk Jelasity, and Vilmos Bilicki. Stunner: A smart phone trace for developing decentralized edge systems. In José Pereira and Laura Ricci, editors, *Proceedings of the 19th International Conference on Distributed Applications and Interoperable Systems (DAIS 2019)*, pages 108–115, Cham, 2019. Springer International Publishing.