I Want to Go Home: Empowering the Lost or Stolen Mobile Device

Chi Zhang
Department of Computer Science and Engineering
Polytechnic Institute of NYU
Brooklyn, NY, 11201
czhang@gmail.com

Robert W.H. Fisher
Department of Machine Learning
Carnegie Mellon University
Pittsburgh, PA 15213
rwfisher@cs.cmu.edu

Joel Wein
Department of Computer Science and Engineering
Polytechnic Institute of NYU
Brooklyn, NY, 11201
wein@poly.edu

ABSTRACT

It is estimated that over 8 million cell phones are lost or stolen each year[**?**]; this problem is particularly acute among academics. Often the loss of a cell phone means the loss of personal data, time and enormous aggravation.

In this paper we discuss preliminary results on machine-learning based algorithms by which a cell phone can discern that it may be lost or stolen, and take steps to enhance its chances of being succesfully recovered. We use data collected from the Reality Mining project, to create a suite of possible test cases that model lost or stolen cell phone behavior. **TODO: Brief clear statement about our results. We can do X with Y accuracy**

Categories and Subject Descriptors

H.4 [**TODO**]: TODO; TODO [**TODO**]: TODO—*TODO1,TODO2*

General Terms

TODO

Keywords

TODO

1. INTRODUCTION

In 2007, Computerworld estimated that over 8 million cell phones would be lost or stolen over the course of that year [**?**]. For many, the loss of their mobile device can be costly, aggravating, and potentially devastating. In the words of one technology blogger:

> Losing your cellphone is like losing your dog. First you panic. Then you spend a lot of time calling it. Then you feel really alone... These devices hold our whole lives. Every old phone number we never memorized, every old photo we looked at daily, every voice mail we wanted to keep. And once the device is lost, so too is our information...

As mobile devices become more powerful, more expensive, and more deeply embedded in our lives, preventing their permanent loss takes on greater urgency. One can take a variety of approaches to the problem, including making data backup easier, or developing methods to send a signal to lost phones[**?**]. In this paper we discuss preliminary results targeted at enabling the mobile device itself to determine that it may be lost and stolen and respond accordingly. We focus on the algorithms that will enable the device to determine that, with high probability, it is lost or stolen.

Our basic approach is to use the toolbox of machine learning; our algorithms allow the mobile device to learn a profile of what is "normal" behavior for the phone, which will then enable it to detect when its behavior becomes "abnormal" and react appropriately. An important and proven application of our techniques is credit card fraud detection. Although accurate aggregate statistics are unusual, researchers predict that nearly one billion dollars are lost to credit card fraud in the United States alone every year[1]. Credit card companies use automated fraud detection algorithms to save hundred of millions of dollars.

We note that in this short paper we do not focus on what the device might do upon deciding that it is lost or stolen. Options range from blaring loud embarrassing messages ("SAVE ME!! SAVE ME!!") to sending out help messages to passing mobile devices, to going into special power-savings modes with periodic wakeups during which the device transmits location or other information –the imaginative reader can surely devise other ideas as well.

We validate our ideas on data sets from the Reality Mining Project [**?**, **?**] which collected over 350,000 hours of cellphone behavior data from Nokia 6600 smartphones. This project, discussed in more detail in Section 2 collected a variety of data (TODO FILL IN) on cell phone usage of approximately 100 users over 9 months. At a high level, our approach is to "train" a machine learning algorithm to learn a profile of mobile device M by giving it some examples of data points that are those of the mobile device M, and some that are not. We then present new data points to the algorithm and ask it to classify whether they seem to be from device M or not.

We believe it is quite feasible to deploy algorithms of the sort we discuss even on relatively computationally limited mobile devices. Traditionally the training phase is some-

HOTMOBILE '10 Annapolis,Maryland

what computationally intensive; however, in a real use case might be carried out at low priority over several days or weeks as the algorithm collects data from the (legitimate) use of the mobile device. Typically, once the classifying algorithm is trained, making classifications of new data points is relatively fast. Thus, a deployed algorithm, once trained, would simply monitor data from the device's usage, and, upon seeing a sequence of examples classified as unusual, raise a flag.

We hope that our preliminary results may inspire discussion in several directions. Most broadly, the amount and quality of data collected by mobile devices is only increasing, and the potential using machine learning to mine this data, in a centralized fashion or by individual units, is interesting and relatively unexplored. [Robert does this last sentence make any sense to you?] [Robert maybe here is where you can first point out that in fields such as credit card analysis, etc. these techniques work well] More narrowly, we seek to share ideas on appropriate test cases on which to try out our algorithms, and discussion on how the more difficult cases may be tackled, and on approaches to further enhance the speed and accuracy of our algorithms. Need more ideas here.

2. DATA SETS

In this section we describe the scenarios on which we test our algorithms.

2.1 Reality Mining Project

The Reality Mining project represents one of the largest publicly available corpuses of cellphone usage data[3]. Collected over the 2004-2005 academic year at the Massachusetts Institute of Technology, the data is comprised of nearly 40 years worth of cell phone activity from 97 volunteer subjects. All subjects used variants of the Nokia Symbian Series 60 phone, and the data includes cellular data activity, cell tower communication, and call logs, including call time, duration, and contact phone number. In addition, all users completed a survey describing information about their personal and professional lives.

We use seven features from the Reality Mining data to represent an individual phone call. Specifically, our algorithm considers:

Day of phone call (i.e. Monday)
Hour of phone call (in the interval [0,23])
Phone Number
Index of contact in phonebook
Call type (SMS/Voice call/Packet Data)
Direction (Incoming/Outgoing/Missed call)
Call duration

2.2 The Stolen Case

Our approach to generating a data set that simulates the "stolen" cellphone is as follows. We assume that the data records would, for some period of time, be drawn from those of one user U1, and then suddenly, perhaps after a brief interlude, be drawn from those of a different user U2. This models user U2 stealing user U1's device and using it as his or her own for a period of time.

To this end we construct the following data sets. As noted previously, users in the database self-classified their behavior as "Predictable", "Somewhat predictable," or "Not Predictable." We chose six users of the 97 from the database, two from each category. Call these P1,P2,S1,S2,N1,N2. For each of these six users we trained an algorithm algorithm to attempt to learn the profile of usage of each user. We will call the trained algorithm for user U A(U). From these 6 users we essentially constructed [CHI: HOW MANY? Is it a total of 15 =(6*5/2) in which you distinguish between each pair is users] test cases constituted by taking a sequence of records from one user and then appending a list of records from a second user. If the algorithm is succesful it will be able to distinguish the appended list from the initial list quite strongly.

2.3 The Lost Case

TODO ALL: I am less sure what is happening here. Here is my understanding.

- Robert had a case where he presented the algorithms just with the sequences of celltowers with which they had contact, and then presented the algorithm with Chi's [random/cycle] sequences of celltowers. I think it makes sense to make this one case.
- CHI TODO: What you wrote about this case in your .doc is not understandable. You need to write 1-2 clear specific and precise paragraphs in as good English as you can muster about EXACTLY what were the data sets you created that simulated lost behavior You took the 6 users P1, P2, S1,S2,N1,N2 and did what? For example:
  - STOLEN DATA SET 1: Identical to regular except outgoing calls removed.
  - STOLEN DATA SET 2: Identical to STOLEN SET 1 except it sits in one place all day
  - STOLEN DATA SET 3: The Bus/Taxi Case? Sequence of cell towers changes

It is important to note that all of these cases may represent legitimate mobile device usage. A user may move, start a new job, travel to a new setting, etc. In these cases the device should support a mode in which the user can tell it, potentially with some associated security measures, that it should not worry, things are fine.

3. OUR ALGORITHMS

3.1 Background on Machine Learning

Broadly speaking, we think of supervised learning as the approximation of some function $f : X \to Y$, where we call X the target function. We are given a collection of data, which consists of input/output pairs (x, y) from the function.

In our work we use Artificial Neural Networks (ANN's) to learn the target function characteristic to lost or stolen cell phones. For each of these two problems, we think of the domain as being phone calls, and the output is a binary variable indicating if the phone call occured while the phone was in the original users possession. Neural Networks are parallel computational models based on the biological brains. The network is represented by a directed graph, $G = (V, E)$, with a weight w_{ij} associated with the edge from node i to node j. A network is constructed of n nodes on the input layer, where n is the number of dimension of the network

inputs, some number of hidden layers, and an output layer with $\log m$ nodes, where m is the number of classes that we are trying to predict. Figure 1 illustrates the structure of an Artificial Neural Network.

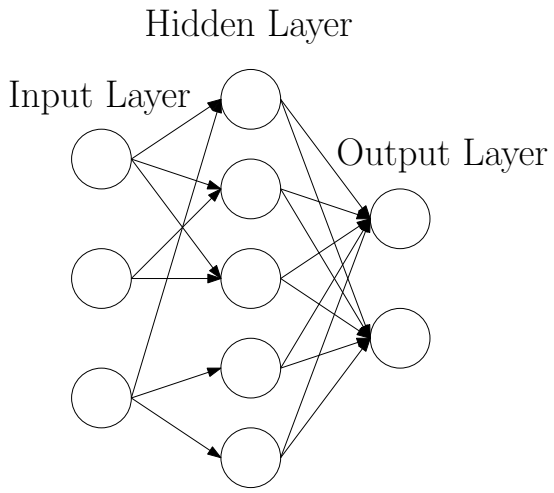

Figure 1: An Artificial Neural Network

An individual neuron, represented by node i in the network, will fire if the following inequality holds:

$$W \leq \sum_j w_{ji} x_j$$

Where $w_{ij=0}$ if there is no edge between nodes i and j in the network. If neuron j fires, then we have $x_j = 1$, otherwise we have $x_j = 0$.

The topology of the neural network, i.e. the number of hidden layers and the placement of the edges, is fixed by the algorithm designer before the training period. The objective of a network training period is to produce values for the weights, w_i, such that the network correctly computes the target function. Most often some form of gradient descent is used to iteratively compute the weights of the network. In our work, we a Neural Network framework known as Levenberg-Marquardt backpropagation[5].

3.2 What We Did

In this section we discuss our algorithms. CHI TODO: Sketch the algorithms you are using as precisely as possible. It should make sense in the context of what Robert wrote. Don't use different names for things. Robert's Background section should make Chi's section understandable.

3.2.1 Algorithms for the Lost Case

3.2.2 Algorithms for the Stolen Case

4. RESULTS

CHI TODO: Sketch your results. Be precise on exactly which experiments you did.

5. RELATED WORK

Neural Networks have been a popular technique for detecting patterns in large, noisy datasets for quite some time. As early as 1993, Neural Network were used in the detection of fraudulent credit card activity [6].

Although we specifically employed the use of Neural Networks in our work, machine learning in general is becoming a popular technique for analyzing cell phone data. The Reality Mining data in particular has been used in dozens of studies, ranging from the prediction of daily user activity[4], to the study of cell tower usage[2].

TODO Please for all of these below provide bibtex references.

- ROBERT TODO: Machine Learning literature on fraud detection and related things
- CHI TODO: Machine Learning literature that develops profiles to optimize power consumption.
- Other approaches to lost cell phones.

6. FUTURE DIRECTIONS

- Discuss how much training data is need and opportunities to optimize – is it practical on a mobile device?
- Discuss computational needs – is it practical on a mobile device?
- How quickly can the phone figure out its lost?

7. ACKNOWLEDGMENTS

Robert Fisher is grateful to the Pittsburgh Chapter of the Achievement Rewards for College Scientists (ARCS) foundation for their support. We also acknowledge Mahadev Satyanarayanan for his useful feedback on our work.

8. REFERENCES

[1] J. R. Dorronsoro, F. Ginel, C. Sgnchez, and C. S. Cruz. Neural fraud detection in credit card operations. *Neural Networks, IEEE Transactions on*, 8(4):827–834, 1997.

[2] N. Eagle, J. A. Quinn, and A. Clauset. Methodologies for continuous cellular tower data analysis. In *Pervasive '09: Proceedings of the 7th International Conference on Pervasive Computing*, pages 342–353, Berlin, Heidelberg, 2009. Springer-Verlag.

[3] N. Eagle and A. Sandy Pentland. Reality mining: sensing complex social systems. *Personal and Ubiquitous Computing*, 10(4):255–268, May 2006.

[4] K. Farrahi and D. G. Perez. What did you do today?: discovering daily routines from large-scale mobile data. In *MM '08: Proceeding of the 16th ACM international conference on Multimedia*, pages 849–852, New York, NY, USA, 2008. ACM.

[5] R. Hecht-Nielsen. Theory of the backpropagation neural network. In *Neural Networks, 1989. IJCNN., International Joint Conference on*, pages 593–605 vol.1, 1989.

[6] S. Maes, K. Tuyls, B. Vanschoenwinkel, and B. Manderick. Credit card fraud detection using bayesian and neural networks. In *Interactive image-guided neurosurgery. American Association Neurological Surgeons*, pages 261–270, 1993.