Towards Safe Cities: A Mobile and Social Networking Approach

Jaime Ballesteros, Bogdan Carbunar, *Member, IEEE*, Mahmudur Rahman, *Member, IEEE*, Naphtali Rishe, *Member, IEEE*, and S.S. Iyengar, *Fellow, IEEE*

Abstract—Population density and natural and man-made disasters make public safety a concern of growing importance. In this paper we aim to enable the vision of smart and safe cities by exploiting mobile and social networking technologies to securely and privately extract, model and embed real-time public safety information into quotidian user experiences. We first propose novel approaches to define location- and user-based safety metrics. We evaluate the ability of existing forecasting techniques to predict future safety values. We introduce iSafe, a privacy-preserving algorithm for computing safety snapshots of co-located mobile devices as well as geosocial network users. We present implementation details of iSafe as both an Android application and a browser plugin that visualizes safety levels of visited locations and browsed geosocial venues. We evaluate iSafe using crime and census data from the Miami-Dade (FL) county as well as data we collected from Yelp, a popular geosocial network.

Index Terms—Context aware safety, distributed algorithms

1 Introduction

Recent technological advances, in particular mobile devices and online social networks, have paved the way toward a smarter management of resources in today's cities. As population density grows and natural disasters and man-made incidents (e.g., hurricanes, earthquakes, riots [1], [2]) impact increasing numbers of people, maintaining the safety of citizens, an essential smart city component, becomes a problem of paramount significance and difficulty.

We envision a system where users are seamlessly made aware of their safety in a personalized manner, through quotidian experiences such as navigation, mobile authentication, choosing a restaurant or finding a place to live. We propose to achieve this vision by introducing a framework for defining public safety. Intuitively, public safety aims to answer the question ''Will location L present any danger for user A when she visits L at a future time T''?

An important challenge to achieving this vision is the need to properly understand and define safety. While safety is naturally location dependent, it is also inherently volatile. It not only exhibits temporal patterns (e.g., function of the season, day of week or time of day) but also depends on the current *context* (e.g., people present, their profile and behavior). Furthermore, as suggested by the above question, public safety has a personal dimension: users of different backgrounds are likely to be impacted differently by the same location/time context.

Previous attempts to make people safety-aware include the use of social media to distribute information about unreported crimes [3], or web-based applications for visualizing unsafe areas [4], [5]. The main drawbacks of these solutions stem from the difficulty of modeling safety and of integrating it in quotidian user experiences.

Instead, in this paper we investigate the combination of space and time indexed crime datasets, with mobile technologies and online social networks to provide personalized and context aware safety recommendations for mobile and social network users. To achieve this, we first define location centric, static crime and safety metrics, based on recorded crime events. Given observed crime periodicities, we show that timeseries forecasting tools are able to predict future crime and safety index values of locations, based on past crime events.

We use statistical tools to show that dependencies exist between the quantity and quality of reviews received by venues in Yelp (a popular geosocial network) and the crime indexes of the venue locations. We then use mobile devices and geosocial networks to record user *trajectory traces*, that enable us to provide personalized, context aware safety recommendations, even when crime information is not available.

We introduce iSafe, a distributed algorithm that addresses privacy concerns raised by the use of trajectory traces and associated crime and safety index values. iSafe takes advantage of the wireless capabilities of mobile devices to compute real-time snapshots of the safety profiles of close-by users in a privacy preserving manner. iSafe uses secret splitting and secure multi-party computation tools to aggregate the trajectories of co-located users without learning the private information of participants.

We have extensively evaluated Android and browser plugin implementations of iSafe, using crime and census data from the Miami-Dade county (FL) as well as data we have collected from the accounts of users and businesses in

- *The authors are with the School of Computing and Information Sciences at the Florida International University, Miami, FL USA. E-mail: {jball008, carbunar, mrahm004, rishen, iyengar}@cs.fiu.edu.*

Manuscript received 16 Feb. 2013; revised 4 July 2013; accepted 24 July 2013. Date of publication 4 Aug. 2013; date of current version 13 Aug. 2014. Recommended for acceptance by J. Lloret.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPDS.2013.190

Yelp [6]. Our conclusion is that iSafe is efficient: even on a smartphone, the computation and communication overheads are a few hundred milliseconds. The iSafe project can be found online [7], providing downloadable Chrome plugin and Android app executables.

The paper is organized as follows. Section 2 presents the system model, the datasets and tools used in this work. Section 3 proposes a static, location centric safety labeling technique and Section 4 compares the ability of existing forecasting tools to predict future crime and safety values. Section 5 introduces the concepts of personalized and context aware safety as well as the iSafe solution. Section 6 investigates relationships between social networks and crime levels. Section 7 describes the iSafe implementation and Section 8 presents evaluation results. Section 9 discusses related work and Section 10 presents our conclusions.

2 Model and Background

We consider a framework consisting of three participants, 1) a service provider, 2) mobile device users, and 3) geosocial networks. The service provider, denoted by S, centralizes crime and census information and provides it upon request. We assume that the mobile devices are equipped with wireless interfaces, enabling the formation of transient, ad hoc connections with neighboring devices. Devices are also equipped with GPS interfaces, allowing them to retrieve their geographic location. Devices have Internet connectivity, which, for the purpose of this work may be intermittent. Users take advantage of Internet connectivity not only to communicate with the geosocial networks but also to retrieve safety information (both described in the following). Each user needs to install an application on her mobile device, which we henceforth denote as the *client*.

Geosocial networks (GSNs) such as Yelp and Foursquare extend classic social networks with the notions of 1) venues, or businesses and 2) *check-ins*. Besides user accounts, GSNs provide accounts also for businesses (e.g., restaurants, yoga classes, and towing companies). GSNs encourage and reward user feedback, in the form of ratings and reviews, left for visited venues. User ratings range from 1 to 5 stars and are aggregated to produce an overall venue rating.

2.1 Data

2.1.1 Geosocial Network Data

We have collected Yelp information from all the venues in the Miami-Dade county, Florida, for a total of 7699 venues. For each venue, we have collected the name, type and address, along with the list of reviews received. For each review, we collected the home city and state of the reviewer. The supplemental material which is available in the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPDS.2013.190 includes plots showing that 1) the number of reviews received by Miami-Dade venues exhibits a long tail distribution and 2) Yelp reviews are mostly positive as most aggregate ratings are at or above 4 stars.

2.1.2 Crime and Census Data

We use a historical database of more than 2.3 million crime incidents reported in the Miami Dade county area since 2007 [8]. Each record is labeled with a crime type (e.g., homicide, larceny, and robbery), the time and the geographic location where it has occurred. We mapped crimes into 7 categories: Murder, Forcible Rape, Aggravated Assault, Robbery, Larceny/Theft, Burglary/Arson, Motor Vehicle Theft. We removed minor crime reports that did not fall into these categories. Let c denote the number of crime types. In our case, $c = 7$. Let $\overline{CT} = \{CT_1, \ldots, CT_c\}$ denote the set of crime types. We also use Census data sets [9], reporting population counts and demographic information. The data is divided into polygon shaped geographical extents called *census block groups*. Each block contains information about the population within (e.g., population count, various statistics). According to the data, Miami Dade county has a population of 2,496,435. The supplemental material available online includes more details of the data classification process and a plot showing the Miami-Dade population density, at block granularity.

2.2 Forecasting and Error Measurement Tools

We rely on time series forecasting tools, including Auto Regressive Integrated Moving Average (ARIMA), Linear (Double) Exponential Smoothing (LES) and Artificial Neural Networks (ANN). The supplemental material available online briefly describes each tool. Furthermore, we use the root mean squared error (RMSE) and mean absolute percent error (MAPE) [10] as error measurement metrics to evaluate the accuracy of the models considered.

2.3 Attacker Model

We consider a semi-honest, or honest-but-curious service provider. That is, the service provider is assumed to follow the protocol correctly, but attempts to learn personal user information as possible. We assume users can be malicious. However, each participating user needs to install a provider-signed client application.

3 Location-Based Safety

We exploit the crime dataset to define an initial, location-centric safety metric. We divide space into census blocks. We divide time into fixed-length epochs, e.g., 1 h long, 24 epochs per day. To understand the need for a time dependent safety metric, we have studied the evolution in time of crimes reported within blocks of the Miami-Dade county. Fig. 1 shows the evolution over seven consecutive days (Wed.-Tue., July 13-19, 2011) of the number of crimes reported within one such block, with a 3 h time granularity. Most of the events are larcenies. The plot shows that the number of crimes reported varies abruptly throughout a day. Case in point, on the depicted Saturday, 7 crimes are reported between hours 15-18, 3 crimes between 18 and 21 and 0 between 21 and 24. Thus, a time-invariant aggregate of past crime events is unlikely to accurately define the present. The supplemental material available online includes

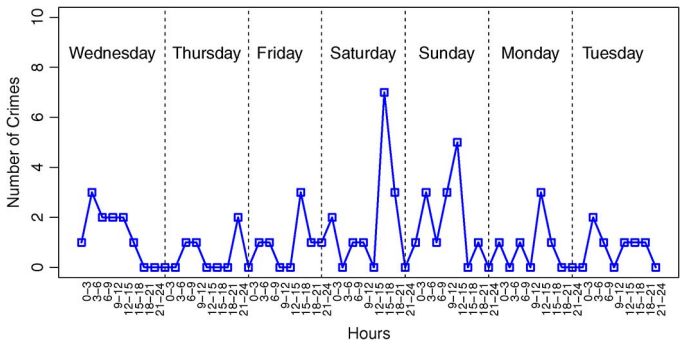

Fig. 1. One-week (July 13-19, 2011) evolution of the number of crimes reported within one Miami-Dade block.

a similar plot, drawn for the same block, over an interval of 18 consecutive weeks.

3.1 Block Crime and Safety Indexes

For a census block B and an epoch e denoted by the time interval ΔT, let $C(B, \Delta T)$ represent a c-dimensional vector, where the i-th entry denotes the number of crimes of type $CT[i]$ recorded in block B during interval ΔT. Let $\overline{W}$ denote a c-dimensional vector of weights; each crime type of $\overline{CT}$ (defined in Section 2.1) has a weight proportional to its seriousness (defined shortly). Let $BC(\Delta T)$ denote the population count recorded for block B. We then define the *crime index* of block B during interval ΔT as

$$CI(B, \Delta T) = \min\left\{\frac{C(B, \Delta T)\overline{W}}{BC(\Delta T)}, 1\right\} \quad (1)$$

where $C(B, \Delta T)\overline{W}$ denotes the vectorial product between the number of crimes per type and the weights of the crime types. That is, B's crime index is the per-capita weighted average of crimes recorded during interval ΔT. The safety index SI of block B during interval ΔT is then defined as

$$SI(B, \Delta T) = 1 - CI(B, \Delta T). \quad (2)$$

Both the CI and SI metrics take values in the [0, 1] interval. In the evaluation section we show that crime index values of blocks in the Miami-Dade county are always smaller than 1. Higher $SI(B, \Delta T)$ values denote safer blocks.

3.2 Crime Weight Assignment

We need to assign meaningful weights to the crime types $\overline{CT}$. An inappropriate assignment may make a large number of ''lighter'' offenses overshadow more serious but less frequent crime events, (e.g., consider larcenies vs. homicides). We propose to assign each crime type a weight proportional to its seriousness, defined according to the criminal punishment code, i.e., the Florida Criminal Punishment Code (FCPC) [11]. The FCPC is divided into *levels* ranging 1-10, and each level L_k contains different types of felonies. The higher the level, the more serious is the felony. Each felony has a *degree*, (i.e., capital, life, first, second and third degree, sorted in decreasing order of seriousness), with an associated punishment (years of imprisonment) [12].

TABLE 1
Crime Weight Assignment Using the FCPC

Crime Type	Weight
Assault	0.176
Robbery	0.180
Rape	0.307
Homicide	0.336

Let L_k denote the set of felonies within level k and let P_k denote the set of corresponding punishments. Let $l_k = |L_k|$ denote the number of felonies within level k. Then, we define the weight of crime type $CT[i]$, $\overline{w}_i$, as

$$\overline{w}_i = \sum_{k=1}^{10} \rho_k \frac{P_k[i]}{\sum_{j=1}^{l_k} P_k[j]},$$

where $\rho_k = k / \sum_{i=1}^{10} i$ is the weight assigned to level k (normalized to the sum of the number of levels). Thus, the weight of crime type $CT[i]$ is the weighted sum of the per-level punishment value ($P_k[i]$) associated with the occurrence of $CT[i]$ within the felonies of level k, normalized to the total punishment of level k. Table 1 shows the resulting weights.

Example. We study the impact of level L_8 on the weight of the ''Robbery'' crime. Out of the felonies represented on level 8, two are related to ''Robbery'': ''Robbery with a weapon'' and ''Home-invasion robbery''. Both are first degree felonies, therefore punishable with up to 30 years of imprisonment. The other represented felonies are ''Homicide'', with 6 different counts, for a total of 135 years penalty and ''Rape'', with 1 count of up to 15 years penalty. Thus, the contribution of level 8 to the weight of ''Robbery'' is $\frac{8}{55} \times \frac{60}{60+135+15} = 0.0415$.

3.3 Illustration

We use the Miami-Dade crime set to illustrate the geographic distribution of block-level safety index information, where the epoch, denoted by the interval ΔT, is the year 2010. We use the census dataset to extract the population count $BC(\Delta T)$. Fig. 2 shows the color-coded safety index for each block group in the Miami-Dade county (FL) where crimes have been reported during 2010. The safety index considers only crimes against persons. Grey blocks have a very low reported crime level. Green blocks denote safer locations while darker yellow and red blocks denote areas with more reported crimes.

4 Predicting Safety

The crime index computation of (1) can only be performed for past epochs, when all crime events have been reported. Safety information however is most useful when provided for the present or near future. One way to predict the crime index of a block B for the next epoch (denoted by the interval ΔT), $PCI(B, \Delta T)$, is the average crime index of the block during the same epoch in the day for the past d days, where d is a system parameter (e.g., $d = 7$ for 1 week of recorded per-block history). This solution however is unable to detect and factor in all crime periodicities, including seasonal, weekly and daily fluctuations. As such, it may include unnecessary errors—e.g., higher number of

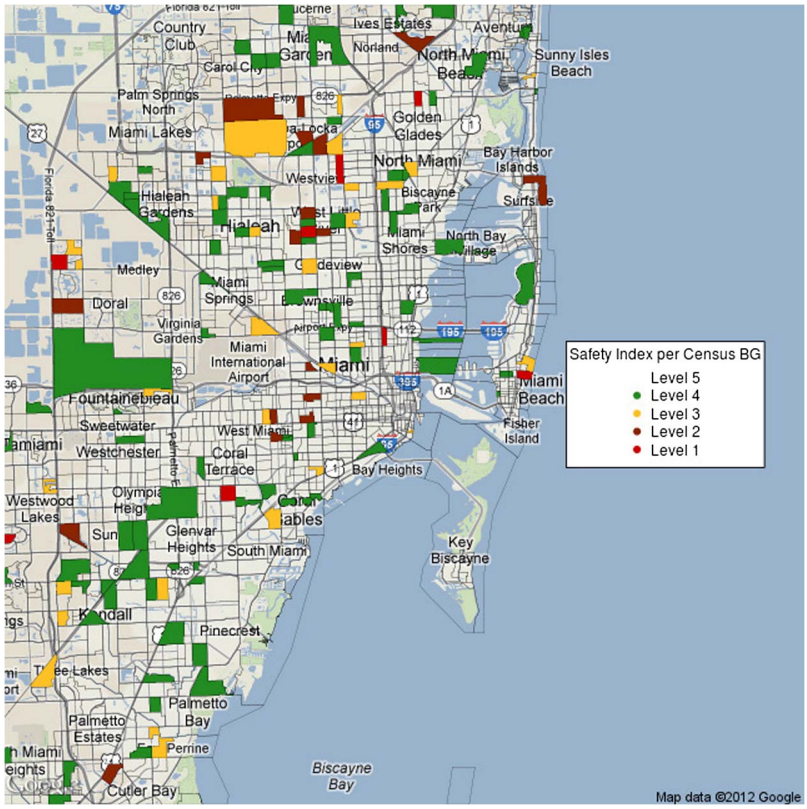

Fig. 2. Safety index illustration for the Miami-Dade county: $SI(B, \Delta T)$ values are mapped into color-coded "safety levels": the higher the level, the safer the block.

crimes in a past August may introduce inaccuracies in the crime index considered in the current month of April.

We propose to address this issue through the use of the time series forecasting techniques discussed in Section 2.2. Specifically, we use time series forecasting tools to compute long and short term predictions of the number of crimes to be committed within an area (e.g., census block, zipcode, and city), based on the area's recorded history. Section 8.2 evaluates the ability of the time series forecasting tools to accurately predict near-future crime counts.

4.1 Predicting Crime and Safety Indexes

At the beginning of each epoch (denoted by the time interval ΔT), compute predictions for the number of crimes of each crime type to be reported at each census block B during the epoch. Let $PC(B, \Delta T)[i]$ denote the predicted number of crimes of type $CT[i]$. Using a formula similar to (1) compute the predicted crime index for B during interval ΔT as $PCI(B, \Delta T) = \min\{PC(B, \Delta T)\overline{W}/BC(\Delta T), 1\}$. The predicted safety index is then $PSI(B, \Delta T) = 1 - PCI(B, \Delta T)$.

5 Personalized, Context-Aware Safety

The ultimate goal of defining crime and safety indexes is to provide users with safety advisory information. People are however not equally exposed and vulnerable to all crime types. Age, gender and an array of personal features, preferences and choices play a central role on the perception of an individual's safety. Since such information may not be readily accessible, we use instead the localization capabilities of a user's mobile device to periodically record and locally store her trajectory trace. This enables us to define the crime index level with which a user is comfortable: the average crime index of the locations in her trajectory. We then introduce personalized safety recommendations both when enough crime information exists to enable the prediction of the near-future crime index of a location and when insufficient such information exists.

We propose to exploit the context of a location, through the people located there. We use the trajectory trace of the user to define the chance of a crime to occur around the user and generalize this approach to compute the chance of a crime to occur around groups of users. This enables us to introduce the concept of *context aware safety*: a user is safe if the chance of a crime to occur around her equals or exceeds the chance of a crime to occur around her co-located users.

5.1 Personalized User Safety

We extend the crime and safety index definitions from locations to users. We assume the device can capture the location of the user with block level precision. Let $TJ_U = \{[B_i, T_i, CI(B_i, \Delta T_i)] | i = 1 \ldots h\}$ denote the trajectory trace of user U, consisting of recorded [block, epoch, crime index] tuples. ΔT_i denotes the epoch containing time T_i, when U was present at block B_i, $T_i \in \Delta T_i$. For privacy reasons, we require each user to store her trajectory trace on her device.

We define the *vicinity crime* metric for a user U, V_U to be the percentage of the user's trajectory places where crimes

have been reported around the time of her visit

$$V_U = \frac{\sum_{i=1}^{h} sgn(CI(B_i, \Delta T_i))}{h} \quad (3)$$

$sgn(x)$ denotes the sign function, that is 0 when x is 0, and 1 when x is larger than 0. For instance, if a user has 100 locations in her trajectory and crimes have been reported at 60 of those locations during the epoch of the user's presence, the user's vicinity crime metric is 60 percent. We then define the crime index of a user U to be the average crime index of locations in her trajectory

$$CI_U = \frac{\sum_{i=1}^{h} CI(B_i, \Delta T_i)}{h}. \quad (4)$$

5.1.1 Safety Decision with Accurate Crime Data

We assume first that user U is located at time T_c in a block B, where accurate past crime data exists. This allows the proper prediction of the crime index, thus the computation of the predicted crime index $PCI(B, \Delta T)$, as specified in Section 4. ΔT denotes the current epoch, $T_c \in \Delta T$. We then introduce the notion of personalized safety recommendation:

Definition 1 (Personalized Safety). *A user U is safe at a block B within time interval ΔT, if $CI_U \geq PCI(B, \Delta T)$.*

Intuition. A user is safe if the user's crime index equals or exceeds the block's crime index predicted for the duration of the user's presence. If the crime index of the user's current block, predicted for the epoch of the user's presence, does not exceed the user's level of comfort, it means the user has spent at least half of her time in locations with more crime than the current location. Thus, the user is likely to be comfortable with the crime level of her current location.

5.1.2 Safety Decision without Accurate Crime Data

Certain locations may have insufficient crime data to ensure an accurate prediction of the location's crime index. For instance, as shown in Fig. 1, the number of recorded events can quickly spike or drop to 0 in short time intervals. Accurately predicting event counts within a short time interval is difficult, as the difference between 0 and 1 crimes is significant. This is the case also during unexpected events (natural and man made disasters) when the future does not reflect the past. To address this issue, we propose to use existing context information, collected from co-located users.

Our approach is the following. We define the safety index of a user U to be the chance of no event being reported in her vicinity: $SI_U = 1 - V_U$. Let $U_1, \ldots, U_k$ be the users co-located with user U. We define a *super user* $SUP_{1\ldots k}$, as a fictitious user whose trajectory trace encompasses the trajectories of users $U_1, \ldots, U_k$. That is, $TJ_{U_{1\ldots k}} = TJ_{U_1} \cup \ldots \cup TJ_{U_k}$. We note that both users and super users can be located in multiple blocks during the same epoch. We then use Equation (3) to compute the vicinity crime metric of $SUP_{1\ldots k}$, $V_{SUP_{1\ldots k}}$. We define the safety index, $SI_{SUP_{1\ldots k}} = 1 - V_{SUP_{1\ldots k}}$. These definitions enable us to introduce the notion of personalized safety recommendation:

Definition 2 (Context-Aware Safety). *A user U is safe in a context consisting of neighboring users $U_1, \ldots, U_k$, if $SI_U \leq SI_{SUP_{1\ldots k}}$, i.e., $V_U \geq V_{SUP_{1\ldots k}}$.*

Thus, a user is safe if surrounded by users whose aggregate safety index is higher or equal to the user's safety index.

Intuition. The safety index of a user encodes the probability that no event occurs around the user. The safety index of a group of users (e.g., $SUP_{1\ldots k}$) is defined as the chance that no event occurs around the group. Definition 2 states that a user is safe if it is surrounded by a group of users whose aggregated chance of no event occurring is higher or equal to the user's chance of no event occurring. A low safety index value does not imply the user is unsafe, but merely the fact that the user spends time in places where events do occur. If the location sampling process is done periodically, the formula naturally ensures that blocks where the user spends more time have more impact on the user's safety index. Being around a group of users whose aggregated safety index is low suggests that the place is likely to have a low safety level.

Factoring in Duration of Stay. The duration of a user's presence within a block needs to be considered when determining the user's safety. For instance, walking through an unsafe block should be avoided. However, when driving on a highway, an unsafe block raises lower safety concerns. One way to address this issue is by using smaller epochs. Another approach is, given a user's trajectory trace, predict the time the user will spend within the current block. The block should raise safety concerns only if the predicted interval exceeds a certain threshold.

5.2 iSafe

User trajectories contain sensitive information, including blocks of interest and behavior patterns. We introduce iSafe, a distributed algorithm that allows the aggregation of trajectory traces of co-located users while preserving the privacy of involved participants. iSafe achieves this by taking advantage of the wireless communication capabilities of user mobile devices to form short lived, ad hoc communities.

5.2.1 Overview

iSafe contacts the neighboring devices, reachable over local wireless interfaces, that run iSafe. If their number exceeds a (system wide) parameter value, iSafe initiates a multiparty computation. The procedure enables iSafe to privately and distributively compute the total number of blocks visited by the owners of those devices as well as the total number of blocks visited that had crimes committed during their presence. This enables iSafe to compute their aggregated vicinity crime index, and rely on Definition 2 to decide the user's safety.

5.2.1 Details

Algorithm 1 contains the pseudocode of iSafe. Its main procedure is $safetyDecision(\Delta T)$, executed periodically by a client C, at C's current block, B. In the first step, C contacts the service provider S, storing the crime and Census datasets. C retrieves the predicted crime index of the block B where the user is located. This operation is performed privately, by using a private information retrieval technique [13]. This prevents S from learning the current location of C.

Algorithm 1 iSafe pseudocode

```
1.Object implementation iSafe;
2.  neighbor[] N;            #set of neighbors
3.  double CI, SI;           #crime, safety indexes
4.  double V;              #vicinity crime prob
5.  BigInteger R;            #random value
6.  BigInteger[]shares;        #set of shares
7.  BigInteger[] NShares;      #shares of neighbors
8.  int BWC;                 #blocks with crime
9.  int TBlk;                 #total blocks visited
10. Operation int safetyDecision(EpochΔT)
11.     B := getCurrentBlock();
12      PCI_B := S.getPCI(B, ΔT);
13.     if (PCI_B! = −1) then return (CI ≥ PCI_B);
14.     else return cas(); fi end
15. Operation int cas()
16.     N := discoverNeighbors();
17.     if (N.size < NThr) then return − 1;
18.     BWC_SUP := multiPartySum(0) − BWC;
19.     TBlk_SUP := multiPartySum(1) − TBlk;
20.     return(V ≥ BWC_SUP/TBlk_SUP); end
21. Operation BigInteger multiPartySum(int type)
22.     R := getRandom();
23.     shares := split(R, N.size);
24.     for i := 1 to N.size do
25.         send(N[i], shares[i]);
26.         NShares[i] := recv(N[i]); od
27.     int order := electLeaderOrder();
28.     BigDecimal S := 0; int count := 0;
29.     while(count < N.size) do
30.         count := count + 1;
31.         if (count = order) then
32.             if (type = 0) then S := S + BWC + R;
33.             else S := S + TBlk + R; fi
34.             for i := 1 to |N| do S := S − NShares[i]; od
35.             mcast(S);
36.         else S := recv(); fi od
37      return S; end
```

If the crime index of the block can be accurately predicted, the operation returns the decision according to Definition 1. Otherwise, it invokes the cas operation. cas first discovers all the ad hoc neighbors of the user. If the number of neighbors is below a system-wide threshold value, $NThr$, it returns -1: not enough information exists to provide an accurate recommendation, and not enough privacy is provided. Otherwise, it invokes the $multiPartySum$ operation twice, with different input arguments. When invoked with argument 0, $multiPartySum$ calculates BWC_{SUP}, the sum of the blocks with crimes visited by all the user's neighbors. When invoked with argument 1, $multiPartySum$ calculates $TBlk_{SUP}$, the sum of the total blocks visited by all the user's neighbors.

The $multiPartySum$ operation is a secure multi-party sum evaluation. It achieves privacy through the use of 1) frequently changing, random MAC addresses for user devices and 2) secret splitting. Each client generates a random value and splits it into shares—one for each neighbor. That is, if the random value is R, the shares $sh_1, \ldots, sh_k$ are generated randomly such that $\sum_{i=1}^{k} sh_i = R$. The client sends each share to one neighbor and receives a share from each neighbor. The clients engage in a leader election and order selection distributed algorithm, where each client is assigned a unique identifier, between 1 and k.

When a client's turn comes, according to the order established, it adds either the user's BWC value (number of census blocks with events visited by the user) or the user's TBlk value (total number of blocks visited), according to the input variable $type$, and adds its random value R to the overall sum (S). It then subtracts all the shares of secrets of its neighbors and sends a multicast of the result, reaching all its neighbors. If it is not the user's turn to transmit, the client waits to receive the multicast values of its neighbors.

The ratio of the computed BWC_{SUP} and $TBlk_{SUP}$ values is the vicinity crime metric of the super user representing the neighbors of C. cas returns the safety decision of Definition 2.

5.3 Analysis

We first define the notion of location privacy in terms of the inability of an adversary $\mathcal{A}$ to guess the location of a user with probability non-negligibly higher than $1/n$, where n is the number of blocks supported by the system.

Definition 3 (Location Privacy). *Let $\mathcal{A}$ control the provider S and any number of clients, such that the number of clients controlled by $\mathcal{A}$ at any location is at most $NThr - h$, where $NThr$ and $h > 1$ are integer parameters. The challenger $\mathcal{C}$ controls one client, Client. $\mathcal{A}$ contacts $\mathcal{C}$ at any time T. $\mathcal{C}$ invokes $safetyDecision(\Delta T)$ on behalf of Client, where B denotes the current block of Client and $T \in \Delta T$. $\mathcal{A}$ outputs B', its guess of the block B where Client is located. We say a solution provides **location privacy** if the advantage of $\mathcal{A}$ in this game, $Adv_{\mathcal{A}} = |Pr[B' = B] - 1/n|$ is negligible.*

We introduce several results whose proofs are included in the supplemental material available online, along with techniques for preventing an adversary from tampering with safety information.

Theorem 1. *An adversary $\mathcal{A}$ controlling $k - h$ out of k participants in the iSafe algorithm, can only find the sum of the input values of the remaining h honest participants.*

Theorem 2. *iSafe provides location privacy.*

An adversary can attempt to use iSafe to identify and target areas considered to be safe. However, safety is personalized: areas denoted ''safe'' for the adversary may not necessarily be safe for other users, who may in effect avoid them. iSafe is also adaptive: newly reported incidents

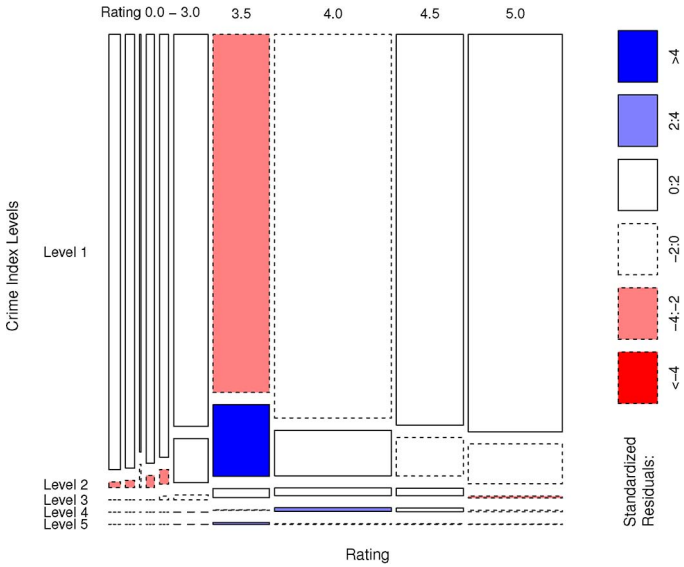

Fig. 3. Mosaic plot showing the relation between venue ratings and the crime index (CI) levels of their location.

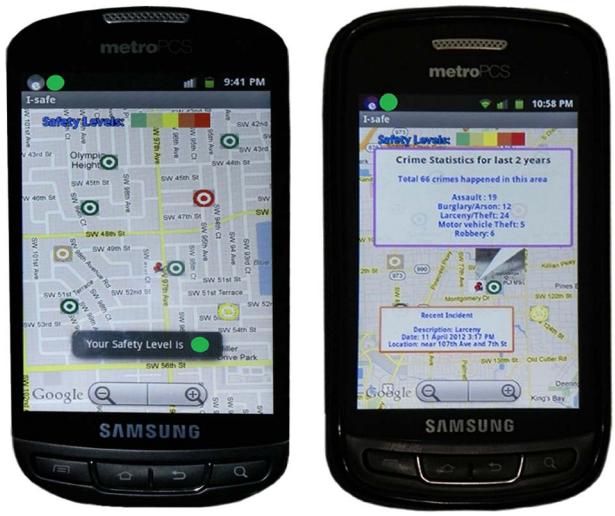

Fig. 4. Snapshots of iSafe on Android.

as well as the lack of incidents are used to continuously adjust block safety values.

6 Geosocial Network Extensions

Geosocial networks seem ideal candidates for augmenting the spatio-temporal context of users. We first investigate relations between crimes and geosocial networking activities. We then propose to use geosocial network user location trajectories to provide safety recommendations.

6.1 Crime vs. Geosocial Activity Dependencies

We conjecture that the crime activity recorded at a location has a bearing on the quality and quantity of reviews recorded at nearby venues. We investigate this hypothesis through the combination of review data we collected from Yelp and the Miami-Dade crime dataset. A first question is whether there exists a relation between the rating of a venue and the safety of its location. For this, we first mapped each venue in the Miami-Dade county to its corresponding census block, then computed CI values for each block using the crime events of 2011. We needed to test for dependencies between two different mixed variables, 1) categorical user ratings and 2) continuous CI values. Since linear regression or any other method for continuous variables are not ideal, we discretized the CI variable into 5 levels, using 1-dimensional k-means (k set to 5), that guarantees optimal partitioning for one-dimensional data.

We have built a contingency matrix, by grouping venues according to their ratings and assigning them to their corresponding CI level: each cell in the contingency matrix contains the number of venues that have the corresponding user rating and belong to a block having the corresponding CI level. We have used the χ^2 test to test the dependency between the two categorical variables [14]. We used the R [15] package to compute the χ^2 test and the p-value (the observed level of significance), and corresponding standard residuals. The standard residuals indicate the importance of the cell to the ultimate χ^2 value. Since the observed level of significance is very close to zero we reject the null hypothesis and conclude that there exists a dependence between CI values and user ratings.

Fig. 3 shows the corresponding mosaic plot, displaying the relationship between ratings and CI values: the areas of the rectangles are proportional to the probabilities of the user ratings and to the conditional probabilities of the CI levels. It shows that the bulk of the Yelp venues (even low rated ones) are in places where crime levels are low.

In the supplemental material available online we confirm the existence of a relation between the number of reviews a venue receives and the safety of the venue's location: Yelp venues with many reviews are located in safer areas than venues with fewer reviews. We also provide an analysis of the dependency sources, through specialized, per crime type views of the data.

6.2 Geosocial iSafe

We extend iSafe with geosocial network information. Specifically, for each geosocial network user U, we define the trajectory trace $TJ_U = \{[B_i, \Delta T, CI(B_i, \Delta T_i)] | i = 1 \ldots h\}$. Each TJ_U record consists of 1) the block containing a venue where U has written a review, 2) the time epoch ΔT when the user wrote the review and 3) the crime index of the block during that epoch. In Yelp, the timestamps associated with reviews have a 1-day granularity, thus, ΔT is 1-day long.

While geosocial network user trajectories are likely to be more sparse than those collected from mobile devices, their similar definition enables us to use Equations (3) and (4) over geosocial trajectories, to compute user vicinity crime metrics and crime index values. These definitions allow us to extend the personalized context aware safety decisions of Section 5.1. Furthermore, the vicinity crime metric and crime index values of users who wrote reviews for a Yelp venue can be used to compute per-venue aggregate crime index and vicinity crime values.

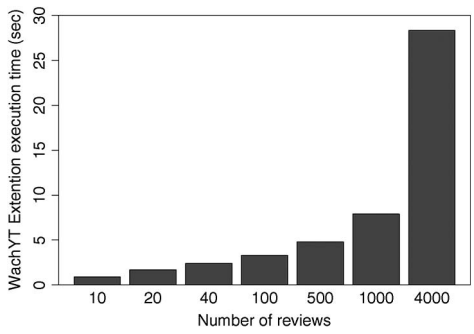

Fig. 5. iSafe browser plugin overhead: Collecting reviews from venues, as a function of the number of reviews.

7 iSafe Implementation

We implemented iSafe as a 1) web server, 2) a browser plugin running in the user's browser, and 3) a mobile app.

7.1 Browser Plugin

We implemented a plugin for the Chrome browser using HTML, CSS and Javascript. The plugin interacts with Yelp pages and the web server, using content scripts (Chrome specific components that let us access the browser's native API) and cross-origin XMLHttpRequests. The plugin becomes active when the user navigates to a Yelp page. For user and venue pages, it parses their HTML files and retrieves their reviews. We employ a stateful approach, where the server's SQLite DB stores all reviews of pages previously accessed by users. This enables significant time savings, as the plugin needs to send to the web server only reviews written after the date of the last user's access to the page.

Given the venue's set of reviews, the server determines the corresponding reviewers. The crime index of blocks of venues reviewed by each user generate the crime index of the user. Crime indexes of reviewers are used to compute the crime index of the venue. The server sends back this information, which the plugin displays in the browser using color codes, ranging from green (safe) to red (unsafe). The supplemental material available online shows a snapshot of the browser plugin.

7.2 Mobile iSafe

We have implemented the location centric static safety labeling component of mobile iSafe using Android. We used the Android Maps API to facilitate the location-based service employed by our approach. iSafe periodically retrieves the user's current GPS location, derives the current census block and also the corresponding crime index. It stores the user's trajectory as one record $[block, time, crime\_index]$ in a local SQLite database. The initial threshold value for creating a new record is 60 seconds.

iSafe uses Bluetooth [16] to compute the vicinity crime metrics of the user's neighbors. We implemented a client-server Bluetooth communication protocol where each device acts as a server and other connected devices act as clients per P2P communication. When compared to Wi-Fi, Bluetooth has drawbacks concerning the transmission range, complexity of the pairing process and the number of communicating peers. However, it also has an important advantage: energy efficiency. Bluetooth consumes less energy than Wi-Fi interfaces, particularly when idle, thus motivating users to leave it always on.

iSafe has a separate background service that displays in the status bar of the Android device, the safety color label of the user's current location. Figs. 4a and 4b show snapshots of the functionality of the mobile iSafe application.

8 Evaluation Results

8.1 Browser Plugin Performance

Fig. 5 shows the overhead of the iSafe plugin when collecting the reviews of a venue browsed by the user, as a function of the number of reviews the venue has. It includes the cost to request each review page, parse and process the data for transfer. It exhibits a sub-linear dependence on the number of reviews of the venue (under 1 s for 10 reviews but under 30 s for 4000 reviews), showing that Yelp's delay for successive requests decreases. While even for 500 reviews the overhead is less than 5 s, we note that this cost is incurred only once per venue. Subsequent accesses to the same venue, by any other user, no longer incur this overhead.

8.2 Forecasting Accuracy

We investigate here the accuracy of the time series forecasting techniques discussed in Section 2.2 in predicting the number of crimes to occur at a location during the near future. We used the R statistical software package [15] to generate the ARIMA model and MATLAB toolboxes [17]

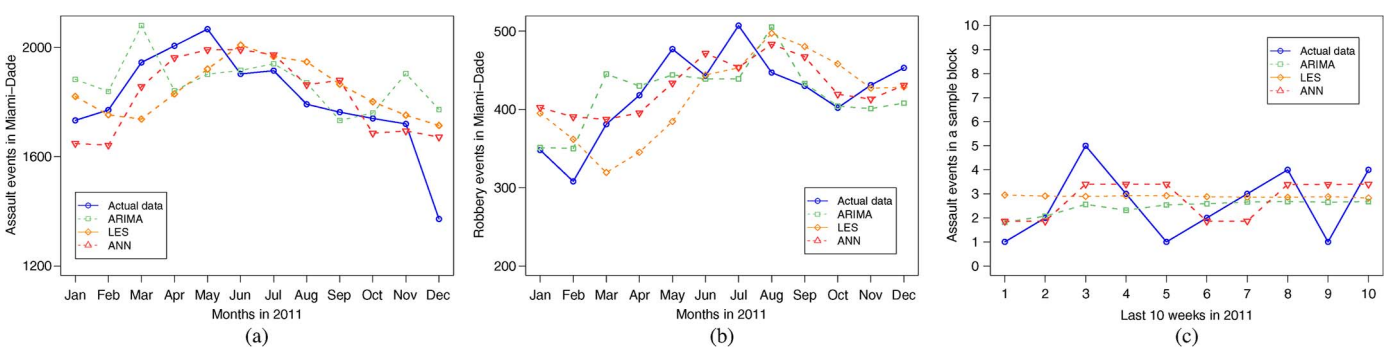

Fig. 6. Crime Forecasting Experiments in Miami-Dade: (a) Prediction of assaults, 2011 monthly basis. (b) Prediction of robberies, 2011 monthly basis. (c) Prediction of assaults in a given block for the last 10 weeks of 2011.

TABLE 2
Error Measurement Data for ARIMA, LES, and ANN. Figures Reference to the Main Document

Model	Figure 5.a		Figure 5.b		Figure 5.c	
	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE
ARIMA	158.80	6.42	38.77	7.08	1.27	43
LES	151.03	6.79	53.57	11.89	1.41	42.08
ANN	116.48	5.32	40.44	8.23	1.3	35.72

for the LES and ANN models. In the following, we analyze separately three crime types: aggravated assault, robbery and larceny/theft that make up for more than 75 percent of the total amount of crimes. For ANN, we set the maximum lag to 12 (to cover the last 12 months/weeks in the lag structure), and the learning rate to 0.1. While a learning rate of 0.4 worked well, we set it to 0.1 to ensure convergence. The higher the learning rate, the faster the network is trained.

We used crime data recorded between 2007 and 2010 to predict per-month categorized event counts for the year 2011, for the Miami-Dade county. Fig. 6a compares the predictions for the number of assaults made by ARIMA, LES and ANN against the recorded values. For ARIMA, we set $\mathrm{p} = 1$, $\mathrm{q} = 1$, $\mathrm{d} = 1$. Details for choosing the ARIMA parameters are provided in supplemental material available online. All three models correctly predict the downward trend from May until December, with ANN achieving a slightly better accuracy than LES and ARIMA. Fig. 6b compares the predictions for the number of robberies. For ARIMA, we set $\mathrm{p} = 3$, $\mathrm{q} = 0$, $\mathrm{d} = 1$. All models accurately predict the initial increase followed by a slight decrease in the number of robberies. ARIMA and ANN outperform the LES model as confirmed by the RSME and MAPE values (see Table 2). ARIMA slightly outperforms ANN.

We further focus on finer grained spatial and temporal predictions: per-block, weekly events. For ANN, we partition the input data into 95 training vectors and 10 test vectors. Fig. 6c compares the recorded data against the ARIMA, LES and ANN predictions of assault events in the last ten weeks of 2011, for one block in the Miami-Dade county. The ARIMA parameters are $\mathrm{p} = 1$, $\mathrm{q} = 1$, $\mathrm{d} = 0$.

8.3 Yelp Safety Profiles

We have collected public information from the accounts of 2025 Yelp users, all residents of the Miami-Dade county. The information collected for each user includes the number of reviews, the venues reviewed, existing check-ins at any venues, and the date when each review and check-in was recorded. We build the crime index, CI, value for each Census block from the Miami-Dade county in 2010. Fig. 7a shows the cumulative distribution function of the CI values (Fig. 2 shows their spatial distribution). It shows that for the Miami-Dade county, most blocks experience relatively low levels of crime per-capita: 50 percent of blocks have a CI value smaller than 0.0015 and only 5 percent of blocks have CI values exceeding 0.01.

Given the CI values of the blocks containing the venues visited (reviewed or subject of a check-in) by a yelper (Yelp user), we compute the user's crime index value, as defined by (4), then the user's safety index: SI_U. Out of the 2025 collected yelpers, 1194 had written reviews in 2010. Fig. 8 shows the distribution of the safety index values of these 1194 yelpers. It shows that most Miami-Dade county yelpers are safe: all have a safety index value larger than 0.96 (1 is the maximum value), with 90 percent of them exceeding 0.99.

We further compare the evolution in time of the safety index SI_B of a block B with the average safety index values over the Yelp users that visited B (and left feedback). To this end, based on the crime database, for each month we calculate the SI value of each block in the Miami-Dade county. We then compute the monthly average of safety index values of yelpers that reviewed venues within B (during the month). Fig. 7b shows the monthly evolution of the SI_B value of a Miami-Dade block and the average safety index value of the Yelp users that visited the block during 2010. For this block, the two metrics have similar values. This shows that an average of the safety indexes of the block's visitors can be used to replace a crime-based safety index for the block.

8.4 Android iSafe Evaluation

We have created a testbed consisting of 4 Android smartphones: Samsung Admire (OS: Gingerbread 2.3.4), HTC Aria (OS: Eclair 2.1), Sony E10i (OS: Eclair 2.1), and Samsung GALAXY S II (OS: Gingerbread 2.3.4). For single device testing, we used the Samsung Admire smartphone with a 800 MHz CPU. Thus, we set the $NThr$ value to 3 and the number of secret shares to 4. In the following, all reported values are averages over at least 10 independent protocol runs.

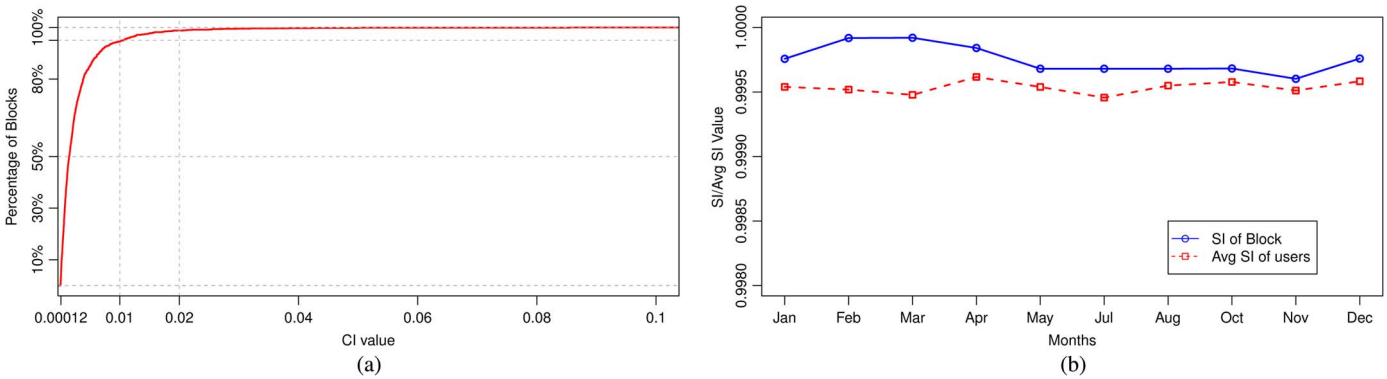

Fig. 7. (a) Distribution of block crime index values in the Miami-Dade county. (b) Evolution in time of the SI value of a Miami-Dade block and the average SI values of Yelp users that visited the block.

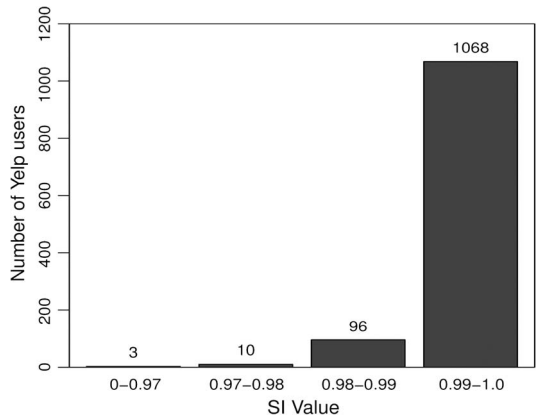

Fig. 8. Distribution of safety index values of Yelp users.

We have first measured the overhead of the secret share generation and reconstruction operation. Fig. 9a shows the overhead on the smartphone, when the modulus size ranges from 64 to 1024 bits. Note that even a resource constrained smartphone takes only 4.5 ms and 16 ms for secret splitting and reconstruction even for 1024 bit long moduli.

Furthermore, we focus on the time and space communication overhead for a single device as well as for the 4 connected devices in our testbed. Fig. 9b shows the dependence of the communication time on the modulus bit size. Even for a modulus size of 1024 bits, the average end-to-end communication overhead of a single device is 342 ms and 1.3 s of our whole system. Fig. 9c shows the dependency of the communication overhead (in KB) on the modulus size ranging from 64 to 1024 bits, for a single device and for the whole system of 4 connected devices. Even for 1024 bit moduli, the total communication overhead is around 3 KB.

9 Related Work

This work extends our initial efforts [18] with additional approach details and evaluations, attacks and defenses, and extensive implementations and evaluations of iSafe including a browser plugin and an Android application.

Smart cities have been the focus of recent efforts at IBM [19] and several academic research groups at MIT [20] and UCLA [21]. Caragliu *et al.* [22] present a study on the factors that determine the performance of a ''smart city''. They focus specifically on European cities by analyzing urban environments, levels of education and different accessibility modalities that are positively correlated with urban wealth. Since one important aspect of smart cities is safety, Patton [23] propose the use of audio sensors and cameras that allow authorities to quickly respond in an emergency event without receiving a 911 call. We note that we consider a preventive angle, of making users aware of their surroundings.

Furtado *et al.* [3] propose the use of social media in a collaborative effort to inform people about crime events that are not reported to police. Their wiki website spots areas on the map where participant users have reported crime events. Police departments also release tools to make citizens aware of their safety, e.g., the Miami-Dade police department, deployed an web application [24] that identifies crime areas based on current crime reports. Instead, iSafe seamlessly integrates context and time sensitive safety metrics into the everyday user experience. Dynamic safety practices leveraging social networks and GPS mobile phones have been introduced in [25] to create a system for personalized safety awareness. The definition of safety indexes that leverage crime, social and mobile activities, as well as the use of safety predictions, differentiate iSafe.

Participatory sensing is receiving increasing attention. Estrin [26] discuss advantages of participatory sensing in health and transportation and provide insights on the architecture of participatory sensing applications. Thiagarajan *et al.* [27] propose cooperative transit tracking using mobile phones. Privacy becomes a serious concern when the user personal information may be compromised. Christin *et al.* [28] present a survey on the efforts made to preserve privacy in participatory sensing systems. In contrast, iSafe does not collect user information, but instead allows devices to aggregate information collected from co-located users without learning personal information.

The problem of crime prediction has been explored in several contexts. Hotspot mapping [29] is a popular analytical technique used by law enforcement agencies to identify future patterns in concentrated crime areas. Different methods and techniques have been analyzed to review the utility of hotspot mapping in [30], [31], [32], [33]. Hot spot analysis however, often lacks a systematic approach, as it depends on human intuition and visual inspection.

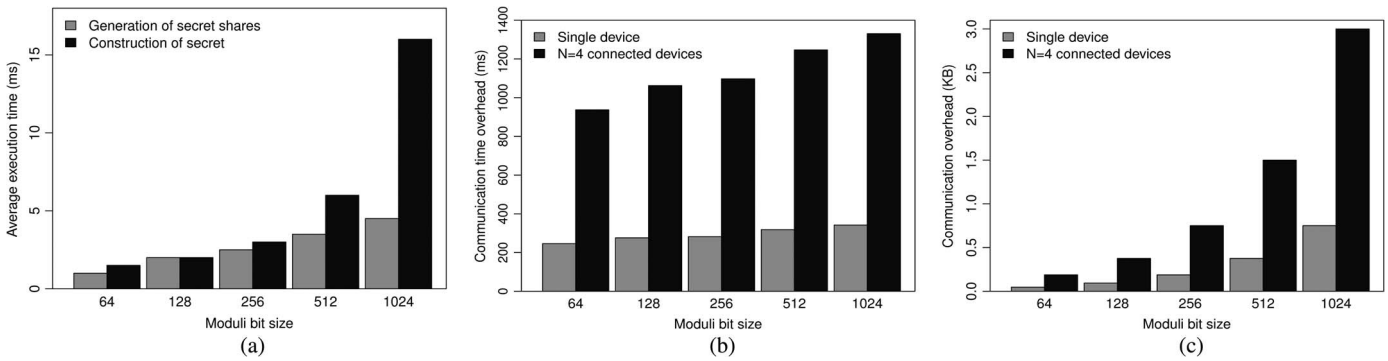

Fig. 9. Android iSafe overhead. (a) Secret share generation and secret reconstruction time overhead. (b) iSafe communication overhead for single device and for all 4 devices. (c) iSafe total communication size for single device and for 4 connected devices.

A variety of univariate and multivariate methods have been used to predict crime. Univariate methods range from simple random walk [34] to more sophisticated models like exponential smoothing. While exponential smoothing offers greater accuracy to forecast ''small to medium-level'' changes in crime [35], we have shown that ARIMA and ANN models outperformed it on our data.

We also note that the end goal of our work is not intrinsically crime forecasting. Instead, we incorporate crime forecasting techniques into our safety metrics, in an attempt to provide to participating users a dynamic framework for safety awareness.

10 Conclusion and Future Work

In this paper we have proposed several techniques for evaluating the safety of users based on their spatial and temporal dimensions. We have shown that data collected by geosocial networks bears relations with crimes. We have proposed a holistic approach toward evaluating the safety of a user, that combines the predicted safety of the user's location with the aggregated safety of the people co-located with the user. Our Android and browser plugin implementations show that our approach is efficient both in terms of the computation and the communication overheads.

In future work we will develop solutions for detecting and eliminating fraudulent information from data sources, including reviews and check-ins. Furthermore, we will integrate safety information in other user experiences, including navigation directions and mobile authentication solutions.

Acknowledgment

This research was supported in part by the Department of Defense under Grant W911NF-13-1-0142 and NSF Grants CNS-1158701, CNS-0963793, CNS-0821345, CNS-1126619, HRD-0833093, IIP-0829576, CNS-1057661, IIS-1052625, CNS-0959985, OISE-1157372, IIP-1237818, IIP-1215201, IIP-1230661, IIP-1026265, IIP-1058606, and IIS-1213026. A preliminary version of this paper appears in IEEE LCN, 2012.

References

[1] ''1992 Los Angeles Riots,'' in *Wikipedia,* Last accessed on July 12, 2012. [Online]. Available: http://en.wikipedia.org/wiki/1992_Los_Angeles_riots
[2] ''2011 England Riots,'' in *Wikipedia,* Last accessed on July 12, 2012. [Online]. Available: http://en.wikipedia.org/wiki/2011_England_riots
[3] V. Furtado, L. Ayres, M. de Oliveira, E. Vasconcelos, C. Caminha, J. D'Orleans, and M. Belchior, ''Collective Intelligence in Law Enforcement: The Wikicrimes System,'' *Inf. Sci.*, vol. 180, no. 1, pp. 4-17, Jan. 2010.
[4] J. Cridland, *Mapping the Riots*. [Online]. Available: http://james.cridland.net/blog/mapping-the-riots/
[5] ''UK Riots: Every Verified Incident,'' in *The Guardian*. [Online]. Available: http://www.guardian.co.uk/news/datablog/2011/aug/09/uk-riots-incident-listed-mapped
[6] Yelp. [Online]. Available: http://www.yelp.com
[7] iSafe: Context Aware Safety. [Online]. Available: http://users.cis.fiu.edu/~mrahm004/isafe/
[8] ''Crimes and Incidents Reported by Miami-Dade County and Municipal Police Departments,'' in *Terrafly Project*. [Online]. Available: http://vn4.cs.fiu.edu/cgi-bin/arquery.cgi?lat=25.81&long=-80.12&category=crime_dade
[9] United States Census, 2010 Census, Washington, DC, USA, 2010. [Online]. Available: http://www.census.gov/2010census/
[10] H. Brian Hwarng and H.T. Ang, ''A Simple Neural Network For ARMA(p, q) Time Series,'' *Omega*, vol. 29, no. 4, pp. 319-333, Aug. 2001.
[11] *Florida Criminal Punishment Code*, Florida Department of Corrections, Tallahassee, FL, USA, 2012. [Online]. Available: http://www.dc.state.fl.us/pub/sen_cpcm/cpc_manual.pdf
[12] R. Hornsby, *Florida Criminal Penalty Chart*. [Online]. Available: http://www.richardhornsby.com/criminal/penalties/
[13] W.I. Gasarch, ''A Survey on Private Information Retrieval (Column: Computational Complexity),'' *Bull. EATCS*, vol. 82, pp. 72-107, 2004.
[14] A.C. Tamhane and D.D. Dunlop, *Statistics and Data Analysis: From Elementary to Intermediate*. Upper Saddle River, NJ, USA: Prentice-Hall, 2000.
[15] *R: A Language and Environment For Statistical Computing*, R Development Core Team, Vienna, Austria, 2011.
[16] *Specification of the Bluetooth System*, Bluetooth SIG, Kirkland, WA, USA, 2001.
[17] *MATLAB*, version 7.10.0 (R2010a), The MathWorks Inc., Natick, MA, USA, 2010.
[18] J. Ballesteros, M. Rahman, B. Carbunar, and N. Rishe, ''Safe Cities. A Participatory Sensing Approach,'' in *Proc. 37th IEEE Int'l Conf. LCN*, 2012, pp. 626-634.
[19] IBM, Armonk, NY, USA, IBM Smarter Cities. [Online]. Available: http://www.ibm.com/smarterplanet/us/en/smarter_cities/overview/index.html
[20] MIT Media Lab, Smart Cities. [Online]. Available: http://cities.media.mit.edu/
[21] Urban Sensing CENS UCLA, Walkability Project. [Online]. Available: https://research.cens.ucla.edu/urbansensing/
[22] A. Caragliu, C. Del Bo, and P. Nijkamp, ''Smart Cities in Europe,'' Faculty Econ., Business Admin. Econom., VU Univ. Amsterdam, Amsterdam, The Netherlands, Serie Res. Memo. 0048, 2009.
[23] Z. Patton, *Sensors Make Cities Smarter*, Apr. 2010. [Online]. Available: http://www.governing.com/topics/public-justice-safety/Sensors-Make-Cities-Smarter.html
[24] Miami-Dade Police Department, CrimeView Community. [Online]. Available: http://gisims2.miamidade.gov/MyNeighborhood
[25] A. Yu, A. Bamis, D. Lymberopoulos, T. Teixeira, and A. Savvides, ''Personalized Awareness and Safety With Mobile Phones as Sources and Sinks,'' in *Proc. Int'l Workshop UrbanSense, Community, Social Appl. Netw. Syst.*, Raleigh, NC, USA, 2008, pp. 26-30.
[26] D.L. Estrin, ''Participatory Sensing: Applications and Architecture,'' in *Proc. 8th Int'l Conf. Mobile Syst., Appl., Serv.*, 2010, pp. 3-4.
[27] A. Thiagarajan, J. Biagioni, T. Gerlich, and J. Eriksson, ''Cooperative Transit Tracking Using Smart-Phones,'' in *Proc. 8th ACM Conf. Embedded Netw. Sensor Syst.*, 2010, pp. 85-98.
[28] D. Christin, A. Reinhardt, S. Kanhere, and M. Hollick, ''A Survey on Privacy in Mobile Participatory Sensing Applications,'' *J. Syst. Softw.*, vol. 84, no. 11, pp. 1928-1946, Nov. 2011.
[29] S. Chaineya, L. Tompson, and S. Uhlig, ''The Utility of Hotspot Mapping for Predicting Spatial Patterns of Crime,'' *Security J.*, vol. 21, no. 1/2, pp. 4-28, Feb./Apr. 2008.
[30] J.E. Eck, S. Chainey, J.G. Cameron, M. Leitner, and R.E. Wilson, ''Mapping Crime: Understanding Hot Spots,'' Special, U.S. Dept. Justice, Office Justice Program, Nat. Inst. Justice, Washington, DC, USA, Aug. 2005.
[31] S. Chainey and J. Ratcliffe, *GIS and Crime Mapping*. Hoboken, NJ, USA: Wiley, 2005.
[32] E. Jefferis, ''A Multi-Method Exploration of Crime Hot Spot: A Summary of Findings,'' U.S. Dept. Justice, Office Justice Program, Nat. Inst. Justice, Washington, DC, USA, Tech. Rep., 1999.
[33] S. Chainey, S. Reid, and N. Stuart, *When is a Hotspot a Hotspot? A Procedure for Creating Statistically Robust Hotspot Maps of Crime*, D. Kidner, G. Higgs, and S. White, Eds. New York, NY, USA: Routledge, 2002.
[34] N. Barberis, A. Shleifer, and R. Vishny, ''A Model of Investor Sentiment,'' *J. Financial Econ.*, vol. 49, no. 3, pp. 307-243, Sept. 1998.
[35] W. Gorr and A. Olligschlaeger, ''Crime Hot Spot Forecasting: Modeling and Comparative Evaluation,'' U.S. Dept. Justice, Office Justice Program, Nat. Inst. Justice, Washington, DC, USA, Draft Final Rep., 2001.

Jaime Ballesteros received the bachelor's degree in computer science from Universidad Javeriana, Colombia. He is now pursuing the PhD degree in the School of Computing and Information Sciences at Florida International University, Miami, FL. His research interests are in large scale data management and data analysis on geographical datasets. In particular, he is exploring algorithms for database joins under composite spatial and textual fuzzy constraints and their applications to geographical data analysis and geosocial networks.

Bogdan Carbunar received the PhD degree in computer science from Purdue University, West Lafayette, IN. He is an Assistant Professor in the School of Computing and Information Sciences at the Florida International University, Miami, FL. Previously, he held various researcher positions within the Applied Research Center at Motorola. His research interests include distributed systems, security and applied cryptography. He is a member of the IEEE.

Mahmudur Rahman received the bachelor's degree in C.S.E from Bangladesh University of Engineering and Technology, Bangladesh and the MS degree in C.S. from Florida International University (FIU), Miami, FL, in 2012. He is pursuing the PhD degree in the School of Computing and Information Sciences, Florida International University, Miami, FL, working under supervision of Dr. Bogdan Carbunar. He spent three years in industry before joining FIU. His research interests are in security and privacy with applications in online and geosocial networks, wireless networks, distributed computing systems and mobile applications. He is particularly interested in studying the tradeoffs between privacy and usability that are achievable in OSNs and strives to provide privacy aware efficient and secure solutions in that context. He is a member of the IEEE.

Naphtali Rishe is the author of three books on database design and geography; editor of five books on database management and high performance computing; owner of four U.S. patents on database querying, semantic database performance, Internet data extraction, and computer medicine; author of 300 papers in journals and proceedings on databases, software engineering, Geographic Information Systems, Internet, and life sciences; awardee of over $45 million in research grants by Government and Industry, including NASA, NSF, IBM, DoI, and USGS; architect of major industrial projects—both prior to his academic career, and as a consultant; founder and director of the High Performance Database Research Center at FIU (HPDRC); director of the NSF Center for Research Excellence in Science and Technology at FIU (CREST) and of the NSF International FIU-FAU-Dubna Industry-University Cooperative Research Center for Advanced Knowledge Enablement (I/UCRC); mentor of 70 postdoctoral, PhD and MS students; and the inaugural FIU Outstanding University Professor. He is a member of the IEEE.

S.S. Iyengar is currently the Ryder Professor and Director of the School of Computing & Information Sciences at the Florida International University, Miami, FL since August 2011 and before he was the Roy Paul Daniels Professor and Chairman of the Computer Science Department at Louisiana State University. During his tenure at LSU he lead the Wireless Sensor Networks Laboratory and the Robotics Research Laboratory. He has also authored/coauthored eight textbooks and edited 12 books in the areas of Distributed Sensor Networks, Parallel Programming and Graph Theory, published in CRC Press/Taylor and Francis/John Wiley/Springer-Verlag/Prentice Hall/Chinese). He is a Fellow of the IEEE, Fellow of Association of Computing Machinery (ACM), American Association for the Advancement of Science (AAAS), Member European Academy of Science (EURASC), and Fellow of The Society for Design and Process Science (SDPS). He is a Golden Core member of the IEEE-CS and a recipient of the Lifetime Achievement Award for Outstanding Contribution to Engineering Awarded by Indian Institute of Technology, Banaras Hindu University, ICAM, 2012. He has published over 400 articles, 40 keynote speeches, yearly workshops at Raytheon, Army Research, and Indo-US Workshop for sensor network, three patents and five patent disclosures pending, supervised 50 PhD dissertations and led the new faculty hiring at FIU and LSU. Further, Iyengar is the founding Editor-In-Chief of the *International Journal of Distributed Sensor Networks* and has been an Associate Editor for *IEEE Transaction on Computers*, *IEEE Transactions on Data and Knowledge Engineering*, and guest Editor of *IEEE Computer Magazine*. He has been an editorial member of many IEEE journals in advisory roles. His research interests include Computational Sensor Networks (Theory and Application) Parallel and Distributed Algorithms and Data Structures Software for Detection of Critical Events Autonomous Systems Distributed Systems Computational Medicine. G-Index: 50 and his papers are cited more than 6000 times.