Towards Safe Cities: A Mobile and Social Networking Approach

Jaime Ballesteros, Bogdan Carbunar, Mahmudur Rahman, Naphtali Rishe, S.S. Iyengar

***Abstract*—Population density, natural and man-made disasters make public safety a problem 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 defining location and user based safety metrics. We evaluate the ability of existing forecasting techniques to predict future safety values. We devise iSafe, a privacy preserving algorithm for computing safety snapshots of co-located mobile devices as well as geosocial network users. We implement iSafe as both an Android application and a browser plugin for visualizing 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.**

I. 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], [3]) impact increasing number 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 of making people safety-aware include the use of social media as a means to distribute information about unreported crimes [4], or web based applications for visualizing unsafe areas [5], [6]. The main drawbacks of these solutions stem from the difficulty of modeling safety and of integrating its use in the everyday life of people. 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.

Specifically, we first define location centric, static crime and safety labels, based on recorded crime events. We take advantage of observed crime behavior periodicities, to conjecture that location safety values are predictable. To verify this hypothesis, we investigate the ability of timeseries forecasting tools to predict future location crime and safety index values based on recorded crime events.

Moreover, we use mobile device and geosocial network technologies to record the *trajectory trace* of a user: the set of (location, time) pairs where the user has been present. When sufficient crime information exists to enable an accurate prediction of location based crime levels, we introduce the concept of *personalized safety* recommendations: A user U is safe at a location L, if the average crime index of the locations in U's trajectory trace equals or exceeds the crime index predicted for the near future at L.

When insufficient crime information exists at a given location, we propose to augment the "context" of the location with data collected from co-located mobile devices and geosocial networks. We define the *vicinity crime probability* metric, to be the chance of crime events being reported around a user or a group of users, based on their past location trajectories. We introduce then the concept of *context aware safety* recommendations: a user U is safe around users $U_1, .., U_k$ if U's vicinity crime probability equals or exceeds the aggregate vicinity crime probability of users $U_1, .., U_k$.

Furthermore, through the statistical χ^2 test we show that dependencies exist between the quantity and quality of reviews venues receive in Yelp (a popular geosocial network) and the crime indexes of the venues's locations. We then propose to similarly augment spatiotemporal context with trajectory traces collected from geosocial network users.

The approach outlined above relies on the ability to aggregate user location trajectories. Access to the trajectory traces of users, along with associated crime and safety index values, either by other users or a centralized service provider, raises significant privacy concerns: even social network providers have been shown to leak [7] and sell [8] user data to third parties.

To address this issue, we devise iSafe, a distributed al-

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

A preliminary version of this paper appears in IEEE LCN 2012.

gorithm that 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 mechanisms to aggregate the trajectories of co-located users without learning the private information of participants.

We have implemented iSafe as a browser plugin component and an Android application. We provide extensive evaluations of our contributions 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 Yelp [9], a popular geosocial network centered on user feedback. Our experiments performed on a testbed consisting of several smartphones show that the Android iSafe app is efficient: the computation overhead is a few milliseconds while the communication overhead is a few hundred milliseconds. The iSafe project can be found online [10], providing downloadable Chrome plugin and Android app executables.

The paper is organized as follows. Section II presents the model considered as well as the datasets and tools used in this work. Section III proposes a static, location centric safety labeling technique and Section IV compares the ability of existing forecasting tools to predict future crime and safety values. Section V introduces the concepts of personalized and context aware safety as well as the iSafe solution. Section VI investigates relationships between social networks and crime levels. Section VII describes the iSafe implementation and Section VIII presents evaluation results. Section IX discusses related work and Section X presents our conclusions.

II. Model and Background

The framework consists of three participants, (i) a service provider, (ii) mobile device users and (iii) geosocial networks. The service provider, denoted by S in the following, centralizes crime and census information and provides it upon request to clients.

We assume 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 report to geosocial networks but also to retrieve crime information (both described in the following). Each user needs to install an application on her mobile device, which we henceforth denote as the *client*.

In the remainder of this section, we describe the geosocial network concept, the crime and census datasets that we use in our work, detail several forecasting tools we use and describe the attacker model we consider in this work.

A. Geosocial Networks

Geosocial networks (GSNs) such as Yelp and Foursquare extend classic social networks with the notions of (i) venues, or businesses and (ii) *check-ins*. Besides user accounts, GSNs provide accounts also for businesses (e.g., restaurants, yoga classes, towing companies, etc). Users “check-in” to report their location, in terms of their presence at one of the venues supported by the GSN. Users can share check-in information with friends and also use it to achieve special status (badges, mayorships) and receive frequent customer discounts from participating venues. In addition, geosocial networks encourage and reward user feedback, in the form of ratings and reviews, left for visited venues. Users rating range from 1 to 5 stars and are aggregated to produce an overall venue rating.

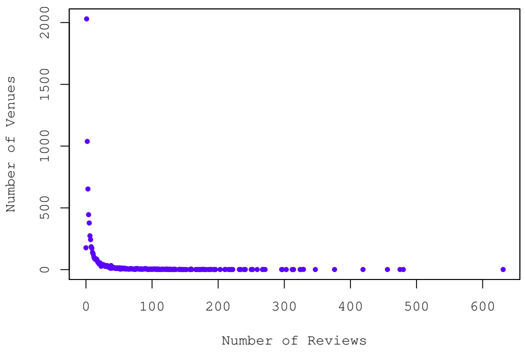


Fig. 1. Miami venue stats: Distribution of number of reviews per venue.

Yelp 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.

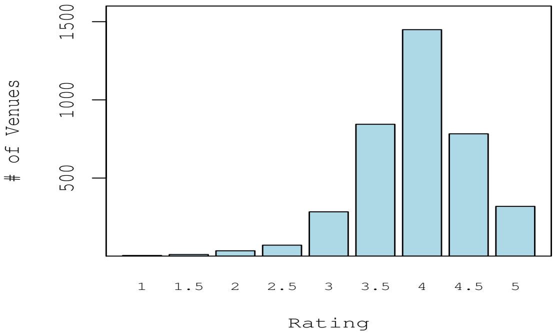


Fig. 2. Miami venue stats: Distribution of venue ratings.

Figure 1 shows the distribution of the per-venue number of reviews of Miami-Dade venues, with a logarithmic y scale. It shows a long tail distribution, with around 2000 venues having 1 review but only 1000 venues having 2 reviews. We emphasize the low number of venues without reviews - only 177. Figure 2 shows the distribution of the number of venues with an aggregated rating ranging between 1 and 5: Yelp reviews are mostly positive as most aggregate ratings are at or above 4 stars.

B. Crime Data

We use a historical database of more than 2.3 million crime incidents reported in the Miami Dade county area since 2007 [11]. Each record is labeled with a crime type (e.g., homicide, larceny, robbery, etc), the time and the geographic location

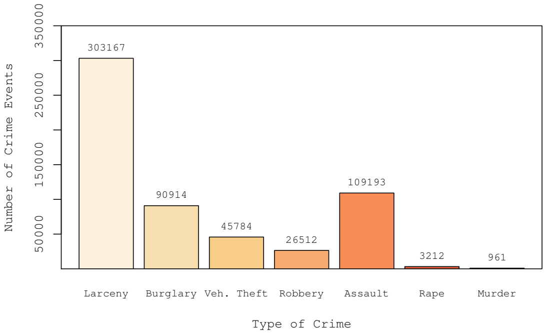


Fig. 3. Outcome of DT classifier – statistics of crime in Miami-Dade county. Distribution of number of crime events per type of crime.

where it has occurred. We briefly document two problems we encountered when pre-processing this data. First, since records come from different Police departments, the crime type labels are non-uniform, (e.g., *murder* in Miami Beach vs. *homicide* in North Miami). Second, crime reports include many minor incidents (e.g., fire alarms issues), resulting in over 140 different crime types.

In order to standardize and eliminate ambiguities, 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. Due to the large number of records in the database, manual mapping was infeasible. Instead, we have experimented with two machine learning techniques for classifying each record: the Naive-Bayes (NB) classifier and the Decision Trees (DT) classifier [12]. In order to build our training and test sets, we manually annotated a random sample of 2000 records from different police departments. Then, we split this subset of records into training and test datasets, each containing 1000 records. We built our classifiers using the NLTK library [13]. The accuracy was measured using a simple metric that measures the percentage of inputs in the test set that the classifier correctly labeled. For instance, a crime type classifier that predicts the correct crime type 60 times in a test dataset containing 100 crime types, would have an accuracy of 60%. On our crime dataset, the NB classifier achieved an accuracy of 91% and the DT classifier an accuracy of 98%. Thus, we have used the outcome of the DT classifier. Figure 3 shows the crime set's distribution of the crime categories following the DT classification.

Let c denotes the number of crime types. In our case, $c = 7$. Let $\overline{CT} = \{CT_1, .., CT_c\}$ denote the set of crime types.

We use Census data sets [14], reporting population counts and demographic information. The data is divided into geographical extents e.g. polygons, 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 $2,496,435$. Figure 4 shows the geographical distribution of the population in the Miami Dade county.

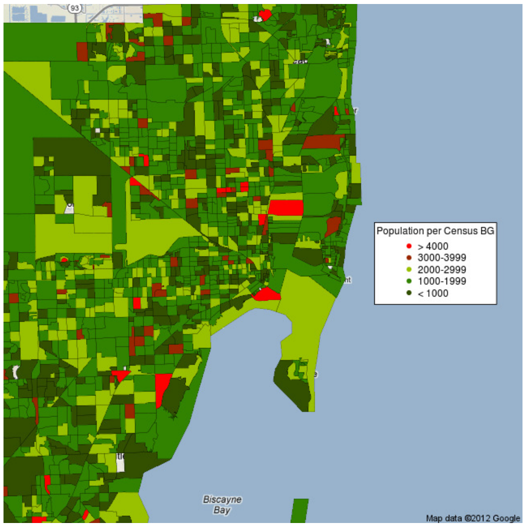


Fig. 4. Miami-Dade county: geographical distribution of population. Polygons represent Census Block Groups.

C. Forecasting Tools

We describe here several time series forecasting tools that we use in our work.

ARIMA Model. ARIMA models incorporate autoregressive (p),integration(d) and moving average terms(q) to provide higher fitting and forecasting accuracy. ARIMA uses the input data to determine the appropriate model form. The ARIMA forecasting procedure consists of four steps [15], (1) identifying the ARIMA(p, d, q) structure, (2) estimating the unknown parameters, (3) fitting tests on the estimated residuals and (4) forecasting future outcomes based on historical data.

Linear (Double) Exponential Smoothing (LES) Model. Brown's linear (double) exponential smoothing [16] includes trend variations of the time series without a significant seasonal component. The process is controlled by a smoothing parameter α whose value ranges between 0 and 1. α decides the weight placed on the most recent observations during the forecast process. We determine the value of α by minimizing the root mean squared error (RMSE) [17] from one step-ahead forecasts and repeating the process for all forecast values.

Artificial Neural Network (ANN). ANNs are data-driven self-adaptive methods that learn and generalize from experience and capture subtle functional relationships among the empirical data even if the inherent relationships are unknown or difficult to describe. In this paper we focus on the multilayer perceptrons (MLP) ANN model, which is particularly suitable for forecasting, due to its ability for input-output mapping. The ANN we consider consists of an input layer (of the same size as the input vector), two layers of hidden nodes and an output layer providing the forecast value. Before the training phase, we normalize the input data to a $(-1, 1)$ range; following the prediction step we map the output back to the initial range. For the training phase we use a multilayer feedforward network trained using back propagation and the Levenberg-Marquardt algorithm to perform function fitting (nonlinear regression).

Error Measurement. We use the root mean squared error

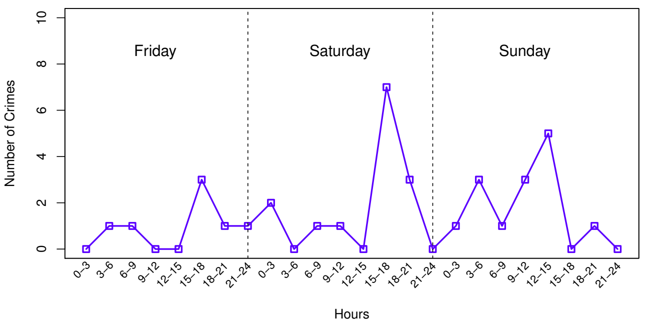


Fig. 5. Three day (July 15-17, 2011) evolution of the number of crimes reported within one Miami-Dade block.

(RMSE) and mean absolute percent error (MAPE) [17] as error measurements to evaluate the accuracy of different models. MAPE can be easily affected by the magnitude of the series but it does provide information about the relative magnitude of the forecast error. On the other hand, RMSE is a more objective measure in absolute magnitude. Thus, in our evaluation, the RMSE is used as the primary and MAPE as the secondary accuracy measure.

D. Attacker Model

We assume 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 as much user information as possible. We assume users can be malicious. However, each participating user needs to install a provider-signed client application.

III. 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 hour 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. Figure 5 shows the evolution over three consecutive days (Friday-Sunday, July 15-17, 2011) of the number of crimes reported within one such block, with a 3 hour time granularity. Most of the events are larcenies. The plot shows a significant variance in the number of crimes reported throughout a day, with a spike between noon and 6pm. Thus, a fixed aggregate of past crime events is unlikely to accurately define the present.

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 II-B) 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

Crime Type	Weight
Assault	0.176
Robbery	0.180
Rape	0.307
Homicide	0.336

TABLE I
Crime weight assignment using the FCPC.

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

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

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

Both the CI and SI metrics take values in the [0, 1] interval. Higher $SI(B, \Delta T)$ values denote safer blocks.

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). Assigning weights to crime types is also a subjective matter: certain people are more likely to be vulnerable to certain crime categories. In the following, we restrict our definition of safety to crimes against persons e.g., assault, robbery, homicide and rape and ignore crimes against property. Although our model can be applied to both categories, the focus of this work is on physical safety.

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) [18]. 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) [19].

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 $\rho_k = k / \sum_{k=1}^{10} k$ is the weight assigned to level k (normalized to the sum of the number of levels). 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 I shows the resulting weights.

Example. We exemplify 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 $\frac{8}{55} \times \frac{60}{60+135+15} = 0.0415$.

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 $BC(\Delta T)$. Figure 6 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. Blocks without color have a very low reported crime level. Green blocks denote safer locations while darker yellow and red blocks denote areas with more reported crimes.

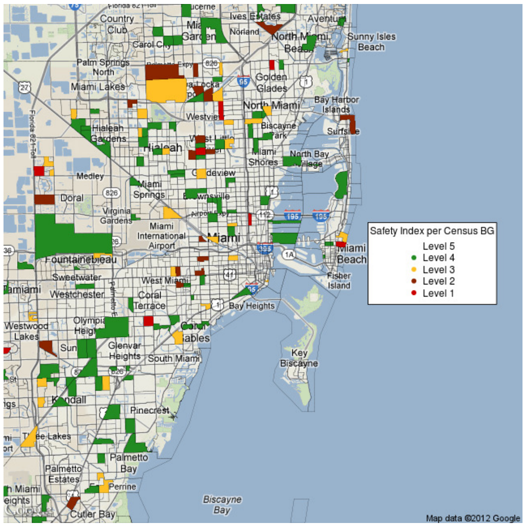


Fig. 6. Safety index illustration for the Miami-Dade county: $SI(B, \Delta T)$ values are mapped into color-coded “safety levels”: the higher the level, the safer the block.

IV. Predicting Safety

The crime index computation of Equation 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 compute the predicted 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 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 II-C. 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, city, etc), based on the area’s recorded history.

Predicting crime and safety indexes. At the beginning of each epoch (denoted by the time interval ΔT), we compute predictions for the number of crimes of each crime type to be committed 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 Equation 1 we 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 $PSI(B, \Delta T) = 1 - PCI(B, \Delta T)$.

V. 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 in 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. When enough crime information exists to enable the prediction of the near-future crime index of a location, we introduce the concept of *personalized safety*: the user is safe if her comfortable crime index level equals or exceeds the predicted crime index of her current location.

However, crime information is not always available or detailed enough to allow a confident prediction of location crime index values. For instance, as shown in Figure 5, the number of recorded events can quickly switch between 0 and 1 in successive intervals. Accurately predicting event counts within a short time interval is difficult, as the difference between 0 and 1 crimes is significant.

We propose to address this issue, by exploiting the intuition that the safety of a place depends not only on its history but also on its current context. One way to define the context of a place at a given time is through the people located there at that time (in Section VI we show how geosocial network data can be used to construct context). We use the trajectory trace of the user to define the probability of a crime to occur around the user and generalize this approach to compute the probability of a crime to occur around groups of users. We then introduce the concept of *context aware safety*: a user is safe if the probability of a crime to occur around her equals or exceeds the probability of a crime to occur around the other users currently co-located with her.

We take advantage of the wireless communication capabilities of user mobile devices to form short lived, ad hoc communities with co-located devices and use them to aggregate the trajectory information of their users. Since user trajectories are sensitive information, we introduce iSafe, a distributed algorithm that allows the aggregation of trajectory traces of co-located users while preserving the privacy of involved participants.

A. Personalized User Safety

We extend the crime and safety index definitions from locations to users. We assume the user's device can capture the user's location, e.g., using GPS or a combination of celltower and Wi-Fi access point localization techniques. We assume a block level localization precision. Let $TJ_U = \{[B_i, T_i, CI(B_i, \Delta T_i)] | i = 1..h\}$ denote the trajectory trace of user U, consisting of recorded [block, epoch, crime index] tuples. ΔT_i denotes the epoch encompassing 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 probability* value of 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 probability is 60%. We then define the crime index of a user U to be the average crime index of locations in her trajectory:

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

1) Safety Decision With Accurate Crime Data: When user U is located at time T_c in a block B, where accurate past crime data exists, allowing the proper prediction of the crime index, we compute the predicted crime index $PCI(B, \Delta T)$, as specified in Section IV, where 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.

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. 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. To achieve this, we exploit ad hoc networks established by devices of co-located users.

Our approach is the following. We define the safety index of a user U to be the probability of no event being reported in her vicinity: $SI_U = 1 - V_U$. Let $U_1, .., U_k$ be the users co-located with user U. We define a *super user* $SUP_{1..k}$, as a fictitious user whose trajectory trace encompasses the trajectories of users $U_1, .., U_k$. That is, $TJ_{U_{1..k}} = TJ_{U_1} \cup .. \cup TJ_{U_k}$. 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 probability of $SUP_{1..k}$, $V_{SUP_{1..k}}$. We define the safety index, $SI_{SUP_{1..k}} = 1 - V_{SUP_{1..k}}$. 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 $U_1, .., U_k$, if $SI_U \leq SI_{SUP_{1..k}}$, i.e., $V_U \geq V_{SUP_{1..k}}$.

That is, the user is safe if it is surrounded by users whose aggregated safety index is higher or equal to the user's safety index.

Intuition. The safety index of a user encodes the probability of no event occurring around the user. The safety index of a group of users (e.g., $SUP_{1..k}$) is defined as the probability of no event occurring around the group. Definition 2 states that a user is safe if it is surrounded by a group of users whose aggregated probability of no event occurring is higher or equal to the user's probability 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.

B. iSafe

One question that remains to be answered is how can the above decisions be made without requiring participating users to provide sensitive location traces and safety index values. To answer this question, we introduce iSafe, a protocol that implements the above solution, in a privacy preserving aware fashion. iSafe consists of a main procedure, $C.safetyDecision(B, \Delta T)$, executed periodically by C, at the C's user current block B.

Definition 3: (Location Privacy)
Let an adversary $\mathcal{A}$ control the service provider S and any number of clients, such that the number of clients controlled by $\mathcal{A}$ at any location is at most $NThr - c$, where $NThr$ and $c > 1$ are integers. The challenger $\mathcal{C}$ controls a client C. $\mathcal{A}$ contacts $\mathcal{C}$ at any time T. $\mathcal{C}$ invokes $C.safetyDecision(B, \Delta T)$, where B denotes C's current block and $T \in \Delta T$. $\mathcal{A}$ outputs B', its guess of the block B where C 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.

Algorithm 1 shows the pseudocode of iSafe. In a first step, the client C installed on the wireless-enabled mobile device of a user 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 (line 12). This operation is performed privately, without the client leaking its location trace, by using a private information retrieval technique [20].

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(Epoch ΔT)
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
15. end
16. Operation int cas()
17.     N := discoverNeighbors();
18.     if (N.size < NThr) then return − 1;
19.     $BWC_{SUP}$ := multiPartySum(0) − BWC;
20.     $TBlk_{SUP}$ := multiPartySum(1) − TBlk;
21.     return(V ≥ $BWC_{SUP}/TBlk_{SUP}$);
22. end
23. Operation BigInteger multiPartySum(int type)
24.     R := getRandom();
25.     shares := split(R, N.size);
26.     for i := 1 to N.size do
27.         send(N[i], shares[i]);
28.         NShares[i] := recv(N[i]); od
29.     int order := electLeaderOrder();
30.     BigDecimal S := 0; int count := 0;
31.     while (count < N.size) do
32.         count := count + 1;
33.         if (count = order) then
34.             if (type = 0) then S := S + BWC + R;
35.             else S := S + TBlk + R; fi
36.             for i := 1 to |N| do S := S − NShares[i]; od
37.             mcast(S);
38.         else S := recv(); fi
39.     od
40      return S;
41. end
```

If the crime index of the block can be accurately predicted (line 13), the operation returns the decision of Definition 1. Otherwise, it invokes the cas operation (line 14). cas first discovers all the ad hoc neighbors of the user (line 17). If the number of neighbors is below a system-wide threshold value, $NThr$, it returns -1: not enough information exists to perform an accurate decision. Otherwise, it invokes the $multiPartySum$ operation twice, with different input arguments (lines 19-20). 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. Thus, the ratio of BWC_{SUP} and $TBlk_{SUP}$ generates the vicinity crime probability of the super user representing the user's neighbors. In line 21, cas returns the safety decision of Definition 2.

The $multiPartySum$ operation is a secure multi-party sum evaluation. It achieves privacy through the use of (i) frequently changing, random MAC addresses for user devices and (ii) secret splitting. Each client generates a random value (line 24) and splits it into shares – one for each neighbor. That is, if the random value is R, the shares $sh_1, .., sh_k$ are generated randomly such that $\sum_{i=1}^{k} sh_i = R$. The client sends each share to one neighbor (lines 26-27) and receives a share from each neighbor (line 28). The clients engage in a leader election and order selection distributed algorithm (line 29), 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), (lines 34-35). It then subtracts all the shares of secrets of its neighbors (line 36) and sends a multicast of the result (line 37), reaching all its neighbors. If it's not the user's transmission turn, the client blocks to receive the multicast values of its neighbors (line 38).

C. Analysis

We now prove the following results.

Theorem 1: An adversary $\mathcal{A}$ controlling $k - c$ out of k participants in the iSafe algorithm, can only find the sum of the input values (BWC or Tblk) of the remaining c honest participants.

Proof: Secret splitting is information theoretical secure: Without knowing *all* the shares of a secret, no information can be inferred about the secret. The adversary $\mathcal{A}$ has access to all intermediate values multicast in Algorithm 1, as well as $k - c$ shares of the secret of each of the remaining c honest participants. Let R_i denotes the random value of the i-th (honest) participant and let $s_{1i}, s_{2i}, .., s_{ki}$ be the shares received by that participant from all the other participants. Then, the sum $R_i + s_{1i} + s_{2i} + .. + s_{ki}$ is random and cannot be predicted by $\mathcal{A}$: $\mathcal{A}$ only controls $k - c$ shares of R_i (out of $k - 1$ shares), but not R_i, thus the other c values in the sum are random and not under the control of $\mathcal{A}$. Thus, $\mathcal{A}$ cannot infer the value (BWC or TBlk) of user i by comparing the value of S before and after user i's multicast. ■

Theorem 2: iSafe provides location privacy.

Proof: (Summary) The adversary $\mathcal{A}$ can only access user location information from (i) user trajectory traces, (ii) queries made by iSafe (Algorithm 1 line 12) and (iii) during computations of the aggregate super user crime and safety indexes (the $multiPartySum$ operation).

For the first point, we observe that user trajectories are only stored on the the user's mobile devices and are never shared with other participants. For the second point, the queries made by users in iSafe to $\mathcal{A}$ are private, e.g., use PIR (see Section V-B). Thus, $\mathcal{A}$ cannot learn the location of the user with a probability non-negligible higher than $1/n$, where n is the number of census blocks, without breaking the security of the PIR solution employed. The third point's implicit requirement is that the provider colludes with users in order to learn information about their neighbors. The use of random, frequently changing MAC (or physical device) addresses by participating devices prevents however even such a powerful adversary from linking a device identifier to a user, thus linking a user to a location. Moreover, Theorem 1 shows that if $\mathcal{A}$ controls at most $NThr - c$ clients at any location where at least $NThr + 1$ clients are located, $\mathcal{A}$ can only learn the sum of the secret values of the remaining (at least c+1, $c > 1$) honest clients. ■

D. Attacks and Defenses

Safety profiles of co-located users are aggregated to obtain a safety image of locations. Since that image impacts user decisions, it can become the target of malicious attacks. For instance, malicious users may attempt to incorrectly (i) improve the safety of desired locations, for instance to attract unsuspecting users to unsafe locations or to (ii) decrease the safety image of target locations. We now describe several mechanisms that could be exploited to perform these attacks, and suggest defenses.

Reporting incorrect locations. Malicious users may report incorrect locations, corresponding to safe areas. Even with GPS verification mechanisms in place, committing location fraud has been largely simplified by the recent emergence of specialized applications for the most popular mobile ecosystems (LocationSpoofer [21] for iPhone and GPSCheat [22] for Android). To prevent this attack, location verification mechanisms can be used [23], [24], [25]. For instance, in previous work [23], one of the authors has developed venue-centric location verification techniques, that rely on devices installed by venue owners within their venues. In the scenario considered in this paper, the owners' incentive for participation is to prevent the tampering of the safety image of their neighborhood.

Turning off devices in unsafe areas. Users could turn off their iSafe application when entering bad areas. While we cannot prevent this behavior, we propose to use rewards and game mechanics to encourage people to report their location. For instance, users gain points for each reported location, perhaps more for the occasional unsafe location. Points are used to acquire badges, similar in principle to those used by geosocial networks like Foursquare [26] or Yelp [9].

VI. Geosocial Network Extensions

Geosocial networks, with their emphasis on the location of both users and venues, seem ideal candidates for augmenting spatiotemporal context. We first investigate relations between crimes and geosocial networking activities. We then propose to use geosocial network user location trajectories to improve the accuracy of iSafe.

A. 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.

One question we need to answer 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 Crime Index (CI) values for each block using the crime events of 2011. We need to test for dependencies between two different mixed variables, (i) categorical user ratings and (ii) 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 then built a contingency matrix, by grouping the 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 [27]. We used the R [28]

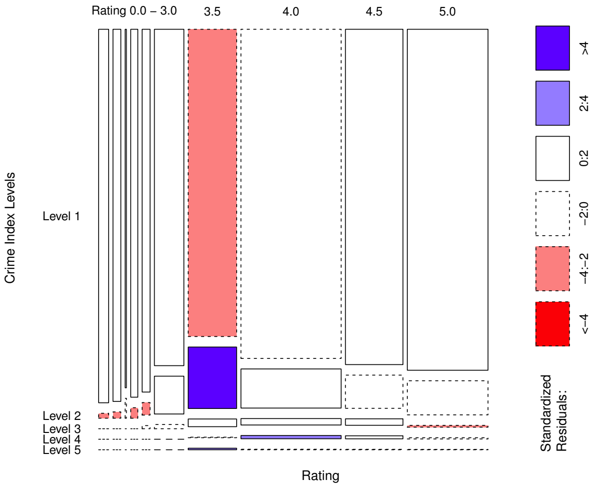


Fig. 7. Mosaic plot showing the relation between venue ratings and the crime index (CI) levels of their location.

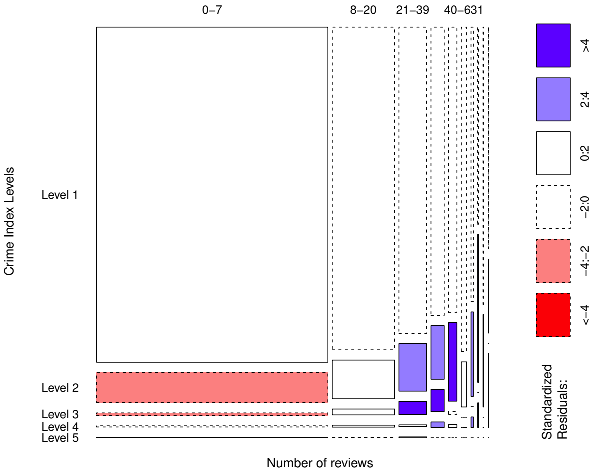


Fig. 8. Mosaic plot showing the relation between the number of reviews received by a venue and the crime index (CI) level of its block.

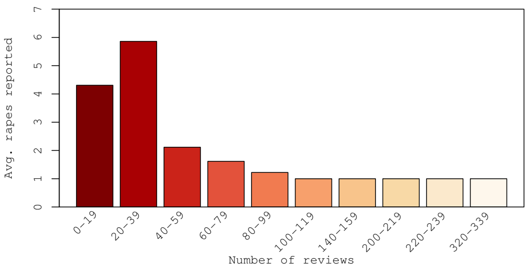


Fig. 9. Number of rapes per number of venue's reviews. Locals and visitors.

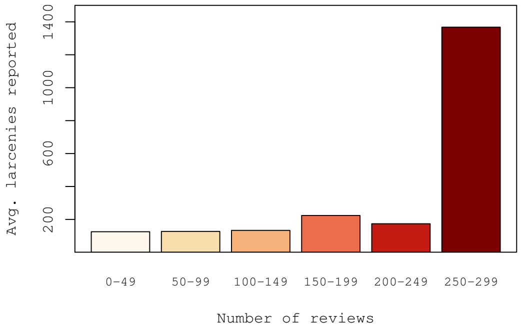


Fig. 10. Number of larcenies/thefts per number of venue's reviews.

package to compute the χ^2 test and we obtained the p-value, or the observed level of significance, and corresponding standard residuals. In short, the standard residuals indicate the importance of the cell to the ultimate χ^2 value; by comparing standard residuals, one can easily identify the cells that contribute the most to the χ^2 test. Since the observed level of significance is extremely low (very close to zero) we reject the null hypothesis and therefore we conclude that there exists a dependence between CI values and user ratings.

Figure 7 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. This can be due to the fact that the distribution of the venues per CI values is long tail, which may be further explained by the fact that (i) in the Miami-Dade county there are few areas with high crime levels and (ii) Yelp is not popular in those areas - people may not even report venues located there in Yelp. Moreover, as shown in Figure 2, Yelp ratings are biased toward higher values.

A second question is whether there exists a relation between the number of reviews a venue receives and the safety of the venue's location. Once again, even though the number of reviews is not a categorical variable, it is discrete. Therefore, we tested their association with CI values using the χ^2 test. We created review count interval buckets and we assigned each venue to one bucket according to its number of reviews. We computed the range of the intervals using the 1-dimensional k-means algorithm with k set to 10. The χ^2 test produced a corresponding p-value very close to zero, thus answering our question in the affirmative. Figure 8 shows the corresponding mosaic plot of this experiment. It confirms that most Yelp venues are located in safe areas as well as the long tail distribution of the number of reviews per venue in Yelp, shown in Figure 1.

In order to identify the sources of the dependencies, we studied a specialized view of this data - the relationship between review counts and crime types (see Section II-B). One finding is depicted in Figure 9, showing the relationship between reported rapes and review counts: rapes occur more frequently in places with low number of reviews. Furthermore, we study the relation between crime types and the number of reviews received from visitors vs. locals. This information is publicly available, as Yelp users need to specify a home city/state. Figure 10 shows that the number of larcenies is high around venues with many local reviews. A potential explanation is that local yelpers (Yelp users) are more likely to choose venues in good neighborhoods, and good neighborhoods are more likely to attract thieves.

B. *Geosocial iSafe*

We propose to extend iSafe with geosocial network information. For each geosocial network user U, we define the trajectory trace $TJ_U = \{[B_i, \Delta T, CI(B_i, \Delta T_i)] | i = 1..h\}$. Each TJ_U record consists of (i) the block containing a venue where U has written a review, (ii) the time epoch ΔT when the user wrote the review and (iii) 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.

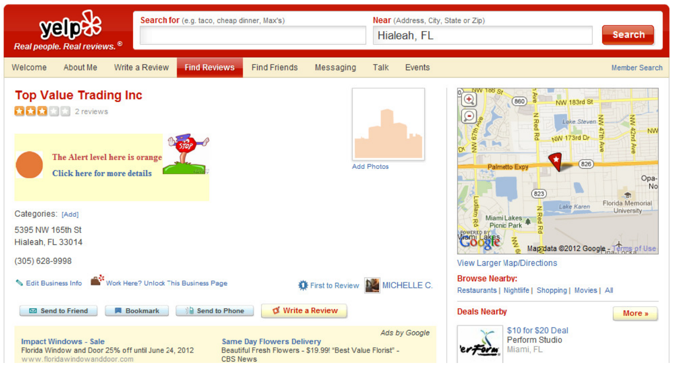
While geosocial network user trajectories are likely to be more sparse than those collected from mobile devices, their similar definition enables us to use Equations 3 and 4 to compute the user's vicinity crime probability and crime index values. Furthermore, we use the vicinity crime probability and crime index values of users who wrote reviews for a Yelp venue to compute aggregate venue crime index and vicinity crime probability values, using the mobile version of iSafe (see Algorithm 1). These definitions allow us to extend the personalized context aware safety decisions of Section V-A.

VII. iSafe Implementation

We implemented iSafe as a (i) web server, (ii) a browser plugin running in the user's browser and (iii) a mobile application. We use Apache Tomcat 6.0.35 to route requests (exposed to the client through a REST API interface) to our server-side component. The server-side component relies on the latest servlet v3.0 which offers additional features including asynchronous support, making the server-side processing much more efficient.

A. *Browser Plugin*

We implemented the browser plugin for the Chrome browser using HTML, CSS and Javascript. The plugin interacts with Yelp pages and the web server, using content scripts (Chrome



yelp Real people. Real reviews.®

Search for (e.g. taco, cheap dinner, Max's)

Near (Address, City, State or Zip) Hialeah, FL

Search

Welcome | About Me | Write a Review | Find Reviews | Find Friends | Messaging | Talk | Events | Member Search

Top Value Trading Inc

2 reviews

The Alert level here is orange

Click here for more details

Add Photos

Categories: [Add]

5395 NW 165th St
Hialeah, FL 33014

(305) 628-9998

Edit Business Info | Work Here? Unlock This Business Page | First to Review | MICHELLE C.

Send to Friend | Bookmark | Send to Phone | Write a Review

Ads by Google

Impact Windows - Sale
Florida Window and Door 25% off until June 24, 2012
www.floridawindowanddoor.com

Same Day Flowers Delivery
Beautiful Fresh Flowers - $19.99! "Best Value Florist" - CBS News

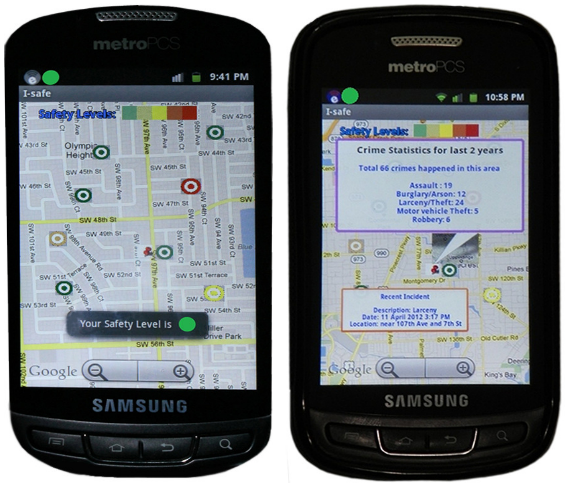
View Larger Map/Directions

Browse Nearby:
Restaurants | Nightlife | Shopping | Movies | All

Deals Nearby | More »

$10 for $20 Deal
Perform Studio
Miami, FL

Fig. 11. Snapshot of iSafe's plugin functionality for a Yelp venue. The orange circle indicates the venue's safety level.



Crime Statistics for last 2 years

Total 66 crimes happened in this area

Assault : 19
Burglary/Arson: 12
Larceny/Theft: 24
Motor vehicle Theft: 5
Robbery: 6

Recent Incident

Description: Larceny
Date: 11 April 2012 3:17 PM
Location: near 107th Ave and 7th St

(a) (b)

Fig. 12. Snapshots of iSafe on Android.

specific components that let us access the browser's native API) and cross-origin XMLHttpRequests. If our content script receives content from another web site, it inspects it for cross-site scripting attacks before injecting the content into the current page (e.g., to protect the user from a hijack attack). To store and process review and user data for each venue, we use the SQLite 3.7.12.1 as the DB server.

The idea behind the browser plugin is to extend the experience of geosocial networks like Yelp [9] with safety information. Specifically, the browser plugin becomes active when the user navigates to a Yelp page. For user and venue pages, the plugin parses their HTML file and retrieves their reviews. We employ a stateful approach, where the server's 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. The initial access is likely to be slower, requiring the plugin to access multiple pages of reviews.

Given the venue's set of reviews, the server determines the corresponding reviewers. Since we do not have access to the location trajectories of users, to compute a user's security label we rely on the venues reviewed by the user: The user safety is computed as an average over the safety labels of the blocks containing the venues reviewed by the user. Given the safety labels of reviewers, we determine the safety level of the venue. The server sends back the safety level of the venue, which the plugin displays in the browser. Figure 11 shows iSafe's extension to the Yelp page of the venue "Top Value Trading Inc." in Hialeah, FL (central left yellow rectangle containing iSafe's safety recommendations).

B. Mobile iSafe.

We have implemented the location centric static safety labeling component of iSafe for a mobile application using Android. We used the Android Maps API to facilitate the location based service employed by our approach. We represent safety using five color labels ranging from green (safe) to red (unsafe).

We used the SQLite version 3.4.0 database to store the trajectory trace of the user, along with timestamps, on her smartphone. The database also caches the Census block structure and associated safety indexes for the city where the user is located. This ensures both (i) privacy – the user trajectory and her requests for block safety indexes never leave her phone and (ii) performance – frequent block safety index requests are performed locally, while infrequent census block safety index updates are performed periodically to ensure an accurate copy of the device's cache.

Whenever a user starts the iSafe app, iSafe retrieves the user's current geolocation, derives the current census block and also the corresponding crime index. iSafe stores the user's trajectory as one record $[block, time, crime\_index]$ in the SQLite database. The initial threshold values for creating a new record are 60 seconds. iSafe uses an exponential backoff algorithm [29] coupled with accelerometer data to ensure that a static device does not consume battery power on GPS queries. iSafe updates then the user's current crime index and vicinity crime probability values.

iSafe uses Bluetooth [30] to compute the vicinity crime probabilities for 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. Bluetooth is a packet-based protocol with a master-slave structure in which one master may communicate with up to 7 slaves in a piconet [30].

iSafe has a separate background service that displays the status bar of the Android device, the safety color label of the user's current location. Figures 12(a) and 12(b) show snapshots of iSafe's functionality.

VIII. Evaluation Results

A. Browser Plugin Performance

Figure 13 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. The experiments were performed on the Dell laptop. It exhibits a sub-linear dependence on the number of reviews of the venue (under 1s for 10 reviews but under 30s for 4000 reviews), showing that Yelp's delay for successive

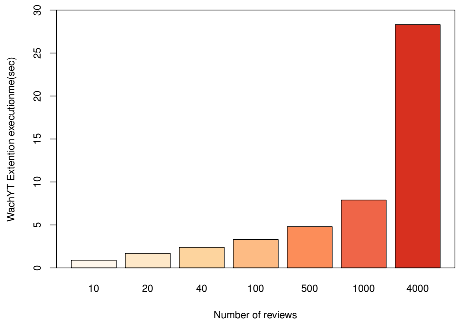


Fig. 13. iSafe browser plugin overhead: Collecting reviews from venues, as a function of the number of reviews.

requests decreases. While even for 500 reviews the overhead is less than 5s, we note that this cost is incurred only once per venue. Subsequent accesses to the same venue, by any other user will no longer incur this overhead.

B. *Forecasting Accuracy*

We explore here the performance of the time series forecasting techniques discussed in Section II-C in predicting the number of crimes to occur at a location during the near future, based on the recorded history.

We used the R statistical software package [28] to generate the ARIMA model and MATLAB toolboxes [31] for 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% of the total amount of crimes. As we show later in this section, predicting categorized event counts enables the prediction of future safety values.

In the first experiment we used crime data recorded between 2007 and 2010 to predict per-month categorized event counts for the year 2011, for the entire Miami-Dade county.

Figure 14(a) compares the predictions for the number of assaults made by ARIMA, LES and ANN against the recorded values. Table II shows the RMSE and MAPE values for the three methods. All three models correctly predict the downward trend from May until December, with ANN achieving a slightly better accuracy than LES and ANN.

Figure 14(b) compares the predictions for the number of robberies made by ARIMA, LES and ANN against the recorded values. 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 II). 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. Figure 15(a) 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. We emphasize the accuracy of the prediction (see Table II), which is similar for ANN and ARIMA. Finally, we focus on daily crime predictions. For the same block used in the previous experiment, using a time window of events recorded between Jan 1, 2010 and Nov 30, 2011, we predict the 31 days of December 2011. Fig 15(b) shows the comparison between the recorded data and the ARIMA, LES and ANN forecast, for the daily number of larceny/theft events.

Experiment conclusions. ANN slightly outperforms ARIMA and LES, but all models exhibit good accuracy - except for the unexpected zero crime incidents observed during a couple of days. Intuitively, using predicted, future values for the number of crimes to define the safety of a block leads to more accurate values than using a static approach.

C. *Yelp Safety Profiles*

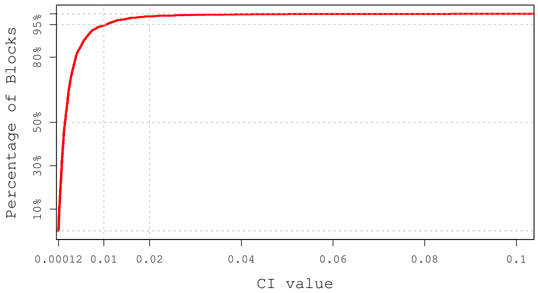


Fig. 16. Distribution of block crime index values in the Miami-Dade county.

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. Figure 16 shows the cumulative distribution function of the CI values (Figure 6 shows their spatial distribution). It shows that for the Miami-Dade county, most blocks experience relatively low levels of crime per-capita: 50% of blocks have a CI value smaller than 0.0015 and only 5% of blocks have CI values exceeding 0.01.

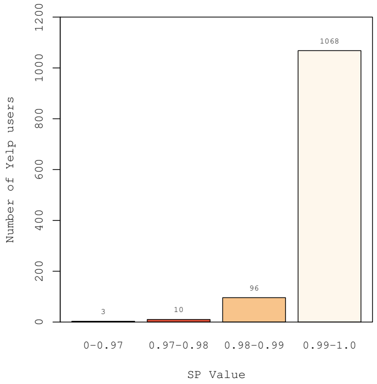
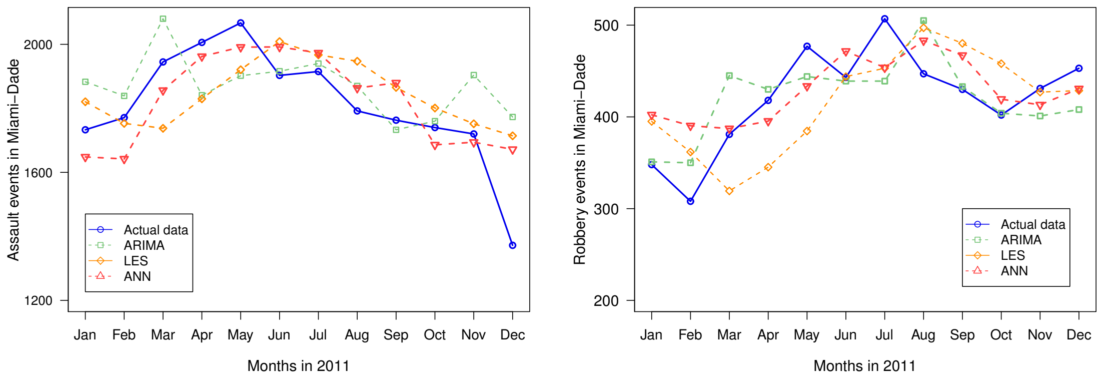


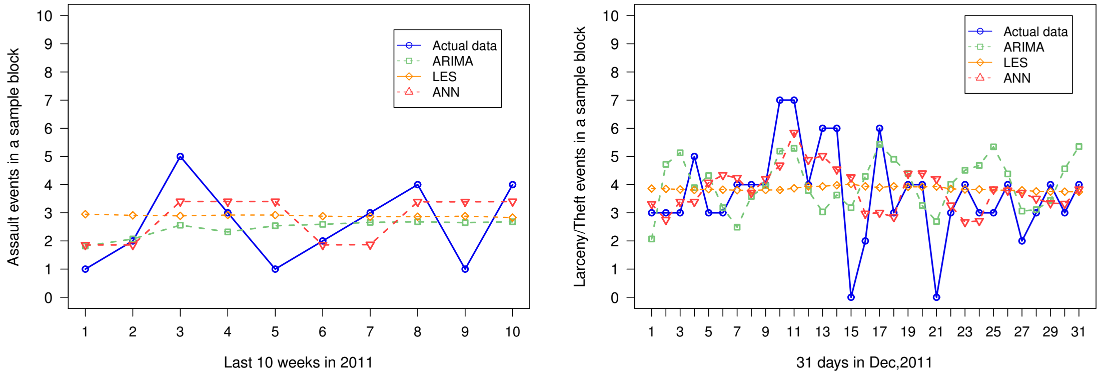
Fig. 17. Distribution of safety index values of Yelp users.

Given the CI values of the blocks containing the venues visited (reviewed or subject of a check-in) by a yelper (Yelp



(a) (b)

Fig. 14. Crime Forecasting Experiments in Miami-Dade: (a) Prediction of assaults, 2011 monthly basis. (b) Prediction of robberies, 2011 monthly basis.



(a) (b)

Fig. 15. Crime Forecasting Experiments in Miami-Dade: (a) Prediction of assaults in a given block for the last 10 weeks of 2011. (b) Prediction of larcenies in a given block for the last 31 days of 2011.

Model	Fig. 14(a)		Fig. 14(b)		Fig. 15(a)		Fig. 15(b)	
	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE
ARIMA	158.80	6.42	38.77	7.08	1.27	43	1.57	34.52
LES	151.03	6.79	53.57	11.89	1.41	42.08	1.61	30.07
ANN	116.48	5.32	40.44	8.23	1.3	35.72	1.49	27.02

TABLE II
ERROR MEASUREMENT DATA FOR ARIMA, LES AND ANN.

user), we compute the user's crime index value, as defined by Equation 4, then the user's safety index: $SI_U = 1 - CI_U$. Out of the 2025 collected yelpers, 1194 had written reviews in 2010. Figure 17 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% 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

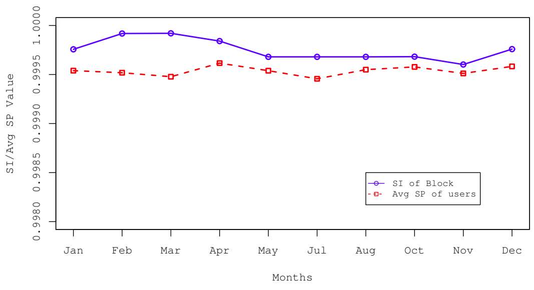


Fig. 18. SI value of a Miami-Dade block and the average of SP values of Yelp users that visited the block w.r.t time.

that reviewed venues within B (during the month). Figure 18 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.

D. 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). We used Shamir's secret sharing solution. For single device testing, we used the Samsung Admire smartphone with a 800MHz CPU. In the following, all reported values are averages taken over at least 10 independent protocol runs.

We have first measured the overhead of the secret share generation and reconstruction operation. Figure 19(a) 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. Figure 19(b) shows the dependence of the communication time on the modulus bit size. Even for modulus size of 1024 bits, the average end-to-end communication overhead of a single device is 342ms and 1.3s of our whole system. Figure 19(c) 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 3KB.

IX. Related Work

This work extends our initial efforts [32] with (i) additional approach details and evaluations, (ii) a list of attacks against our solutions and defenses provided and (iii) 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 [33] and several academic research groups at MIT [34] and UCLA [35]. Caragliu et. al. [36] 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 [37] emphasizes 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 different angle: making users aware of their surroundings.

Furtado et. al. [4] 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 [38] that identifies crime areas based on current crime reports. We note however that our solution seamlessly integrates context and time sensitive safety metrics into the everyday user experience.

Participatory sensing is receiving increasing attention due to the popularity of mobile devices. The multimodal sensing capabilities of devices enable a broad range of applications that leverage collected data from participants, sensed from their surroundings. Estrin [39] discuss advantages of participatory sensing in health and transportation and provide insights on the architecture of participatory sensing applications. Thiagarajan et. al. [40] propose cooperative transit tracking using mobile phones. Privacy becomes a serious concern when the user personal information may be compromised. Christin et. al. [41] present a survey on the efforts made to preserve privacy in participatory sensing systems. In contrast, our work does not collect user information, but instead allows devices to aggregate information collected from co-located users without learning personal information.

Dynamic safety practices leveraging social networks and GPS mobile phones have been introduced in [42] to create a system for personalized safety awareness. The system exploits sensors available in mobile phones to enhance the personal safety of users by aggregating community. Our work is different in that we predict future crime levels, define a safety index that includes the impact of crimes on locations and on the profiles of users and propose a distributed algorithm that privately aggregates safety indexes of co-located users.

The problem of crime prediction has been explored in several contexts. Hotspot mapping [43] 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 [44], [45], [46], [47]. Hot spot analysis however, often lacks a systematic approach, as it depends on human intuition and visual inspection.

A variety of univariate and multivariate methods have been used to predict crime. Univariate methods range from simple random walk [48] to more sophisticated models like exponential smoothing. While exponential smoothing offers greater accuracy to forecast "small to medium-level" changes in crime [49], we have shown that ARIMA and ANN models outperformed it on our data. In [50], Ediger et al. show the effectiveness and reliability of ARIMA and SARIMA models in predicting the total primary energy demand of Turkey from 2005 to 2020. Olligschlaeger [51] showed that ANNs were able to predict drug markets. We note that the 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.

X. Conclusions

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

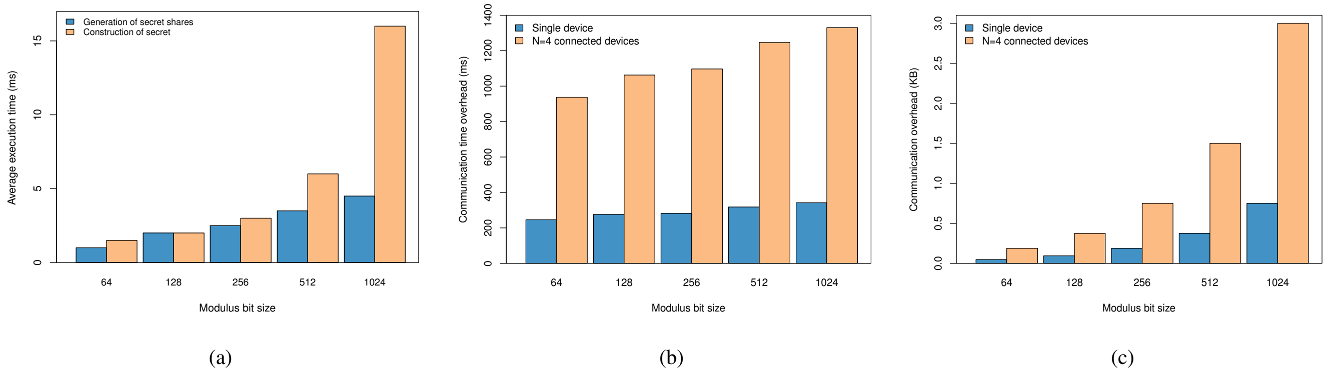


Fig. 19. Android iSafe overhead. (a) Secret share generation and secret reconstruction time overhead. (b) iSafe communication overhead for single device and for all 4 devices. (c) iSafe total communication size for single device and for 4 connected devices.

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.

References

[1] 1992 Los Angeles riots. Wikipedia, at http://en.wikipedia.org/wiki/1992_Los_Angeles_riots, Last accessed on July 12, 2012.

[2] 2005 civil unrest in France. Wkipedia, at http://en.wikipedia.org/wiki/2005_civil_unrest_in_France, Last accessed on July 12, 2012.

[3] 2011 England riots. Wikipedia, at http://en.wikipedia.org/wiki/2011_England_riots, Last accessed on July 12, 2012.

[4] Vasco Furtado, Leonardo Ayres, Marcos de Oliveira, Eurico Vasconcelos, Carlos Caminha, Johnatas DOrleans, and Mairon Belchior. Collective intelligence in law enforcement the wikicrimes system. *Information Sciences*, 180(1):4 – 17, 2010.

[5] James Cridland. Mapping the riots. http://james.cridland.net/blog/mapping-the-riots/.

[6] The Guardian. Uk riots: every verified incident. http://www.guardian.co.uk/news/datablog/2011/aug/09/uk-riots-incident-listed-mapped.

[7] Balachander Krishnamurthy and Craig E. Wills. On the leakage of personally identifiable information via online social networks. *Computer Communication Review*, 40(1):112–117, 2010.

[8] Emily Steel and Geoffrey Fowler. Facebook in privacy breach. http://online.wsj.com/article/SB10001424052702304772804575558484075236968.html.

[9] Yelp. http://www.yelp.com.

[10] iSafe: Context Aware Safety. http://users.cis.fiu.edu/∼mrahm004/isafe/.

[11] Terrafly Project. Crimes and Incidents Reported by Miami-Dade County and Municipal Police Departments. http://vn4.cs.fiu.edu/cgi-bin/arquery.cgi?lat=25.81&long=-80.12&category=crime_dade.

[12] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Addison Wesley, 1 edition, May 2005.

[13] NLTK Project. Natural Language Toolkit. http://nltk.org/.

[14] United States Census. 2010 census. http://2010.census.gov/2010census/, 2010.

[15] Fang-Mei Tseng and Gwo-Hshiung Tzeng. A fuzzy seasonal arima model for forecasting. *Security Journal*, 126:367 – 376, 2002.

[16] Robert F. Nau. Decision 411 forecasting. http://www.duke.edu/ rnau/411avg.htm.

[17] H.Brian Hwarng and H.T Ang. A simple neural network for arma(p,q) time series. *Omega*, 29(4):319 – 333, 2001.

[18] Florida Department of Corrections. Florida criminal punishment code. http://www.dc.state.fl.us/pub/sen_cpcm/cpc_manual.pdf.

[19] Richard Hornsby. Florida criminal penalty chart. http://www.richardhornsby.com/criminal/penalties/.

[20] William I. Gasarch. A survey on private information retrieval (column: Computational complexity). *Bulletin of the EATCS*, 82:72–107, 2004.

[21] Big Boss. Location spoofer. http://goo.gl/59HMk, 2011.

[22] Gpscheat! http://www.gpscheat.com/.

[23] Bogdan Carbunar and Rahul Potharaju. You unlocked the Mt. Everest Badge on Foursquare! Countering Location Fraud in GeoSocial Networks. In *To appear in Proceedings of the 9th IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS)*, 2012.

[24] Stefan Saroiu and Alec Wolman. Enabling New Mobile Applications with Location Proofs. In *Proceedings of HotMobile*, 2009.

[25] Z. Zhu and G. Cao. APPLAUS: A Privacy-Preserving Location Proof Updating System for Location-based Services. In *Proceedings of IEEE INFOCOM*, 2011.

[26] Foursquare. https://foursquare.com/.

[27] A. C. Tamhane and D. D Dunlop. *Statistics and data analysis: From elementary to intermediate*. Upper Saddle River, NJ: Prentice Hall, 2000.

[28] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2011. ISBN 3-900051-07-0.

[29] Byung-Jae Kwak, Nah-Oak Song, and Leonard E. Miller. Performance analysis of exponential backoff. *IEEE/ACM Transactions on Networking*, 13(2), April 2005.

[30] Bluetooth SIG. Specification of the bluetooth system, 2001.

[31] MATLAB. *version 7.10.0 (R2010a)*. The MathWorks Inc., Natick, Massachusetts, 2010.

[32] Jaime Ballesteros, Mahmudur Rahman, Bogdan Carbunar, and Naphtali Rishe. Safe Cities. A Participatory Sensing Approach. In *To appear in Proceedings of the 37th IEEE International Conference on Local Computer Networks (LCN)*, 2012.

[33] IBM. Ibm smarter cities. http://www.ibm.com/smarterplanet/us/en/smarter_cities/overview/index.html.

[34] MIT Media Lab. Smart cities. http://cities.media.mit.edu/.

[35] Urban Sensing CENS UCLA. Walkability project. http://urban.cens.ucla.edu/projects/walkability/.

[36] A. Caragliu, C. Del Bo, and P. Nijkamp. Smart cities in europe. Serie Research Memoranda 0048, VU University Amsterdam, Faculty of Economics, Business Administration and Econometrics, 2009.

[37] Z. Patton. Sensors make cities smarter. http://www.governing.com/topics/public-justice-safety/Sensors-Make-Cities-Smarter.html, April 2010.

[38] Miami-Dade Police Department. Crimeview community. http://crimemaps.miamidade.gov.

[39] Deborah L. Estrin. Participatory sensing: applications and architecture. In *Proceedings of the 8th international conference on Mobile systems, applications, and services*, 2010.

[40] A. Thiagarajan, J. Biagioni, T. Gerlich, and J. Eriksson. Cooperative transit tracking using smart-phones. In *8th ACM Conference on Embedded Networked Sensor Systems*, pages 85–98, 2010.

[41] D. Christin, A. Reinhardt, S. Kanhere, and M. Hollick. A survey on privacy in mobile participatory sensing applications. *Journal of Systems and Software*, 84(11):1928 – 1946, 2011.

[42] Anna Yu, Athanasios Bamis, Dimitrios Lymberopoulos, Thiago Teixeira, and Andreas Savvides. Personalized Awareness and safety with mobile phones as sources and sinks. In *International Workshop on Urban, Community, and Social Applications of Networked Sensing Systems (UrbanSense08)*, 2008.

[43] Spencer Chaineya, Lisa Tompson, and Sebastian Uhlig. The utility of hotspot mapping for predicting spatial patterns of crime. *Security Journal*, 21:4 – 28, 2008.

[44] John E. Eck, Spencer Chainey, James G. Cameron, Michael Leitner, and Ronald E. Wilson. Mapping crime: Understanding hot spots. Special, U.S. Department of Justice, Office of Justice Program, National Institute of Justice, August 2005.

[45] S. Chainey and J. Ratcliffe. *GIS and Crime Mapping*. Wiley, 2005.

[46] E. Jefferis. A multi-method exploration of crime hot spot: A summary of findings. Technical report, U.S. Department of Justice, Office of Justice Program, National Institute of Justice, 1999.

[47] S Chainey, S Reid, and N Stuart. *When is a Hotspot a Hotspot? A Procedure for Creating Statistically Robust Hotspot Maps of Crime*. Kidner, D and Higgs, G and White, S, 2002.

[48] N. Barberis, A. Shleifer, and R Vishny. A model of investor sentiment. *Journal of Financial Economics*, 49:307–243, 1998.

[49] Gorr and A. Olligschlaeger. Crime hot spot forecasting: Modeling and comparative evaluation. Draft final report, U.S. Department of Justice, Office of Justice Program, National Institute of Justice, 2001.

[50] Volkan S. Ediger and Sertac Akar. Arima forecasting of primary energy demand by fuel in turkey. *Energy Policy*, 35:1701–1708, 2007.

[51] A.M Olligschlaeger. Artificial neural networks and crime mapping. Studies/research report, U.S. Department of Justice, Office of Justice Program, Drug Market Analysis Program, 1997.