INFERRING SENSITIVE INFORMATION FROM SEEMINGLY BENEVOLENT SMARTPHONE DATA

Anthony Quattrone (University of Melbourne)
Supervisors: PROF Lars Kulik, **A/PROF Egemen Tanin** (University of Melbourne)

Presented by Anthony Quattrone

Mobile Smartphones

- Mobile smartphones have become ubiquitous
- Success of mobile technology has led to a strong market for the following products and services:
  - Third Party Apps (Facebook, WhatsApp, Shazam)
  - Cloud Storage Providers (Amazon AWS, Microsoft Azure)
  - Location Based Services (Google Maps, Open Street Map)
  - Real-Time Sharing Services (Uber, UberEATS)
  - Wearables (Fitbit, Microsoft Band)

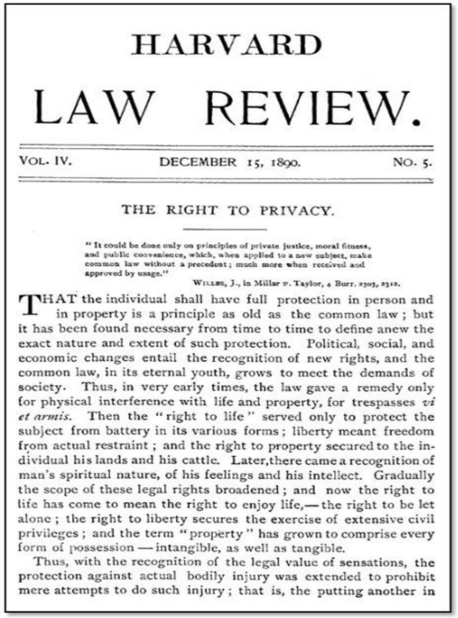
- A mobile device captures more personal information about a user than any other device they own
- **Sensitive mobile information can be easily accessed via standard developer APIs**
- **Literature to highlight potential privacy attacks is scarce**

Seemingly Benevolent Data?

- The primary aim of the research is to determine if data that appears to be benevolent reveals sensitive insights upon further inspection
- Throughout this work we discovered that:
  - Spatial **query results** can be used to reconstruct actual trajectories
  - Bluetooth beacons collecting **signal strength data** can reveal context
  - **Signal strength data** can be used to locate people indoors
  - Encounters between individuals can be detected using **continuous location updates** now commonly provided by popular smartphone platforms
  - **Diagnostic data** and **user settings information** commonly sent in bug reports is unique enough to identify users
- The secondary aim is to safeguard users against such attacks. We developed PrivacyPalisade for the Android platform

Foundations of Privacy

- The **Right to Privacy** published in 1890 was inspired by issues of general coverage of people's personal lives in newspapers
- At the time, the law did not protect people from privacy inferences from the press, photographers or any other modern recording devices
- The article is considered by law scholars to be the foundations of many modern privacy laws
- Information Technology has since advanced considerably with the advent of
  - Database Technology
  - Desktop Computers
  - Internet
  - **Smartphones**
- **Privacy concerns historically have continued to arise which has been the subject of much research**



HARVARD

LAW REVIEW.

VOL. IV. DECEMBER 15, 1890. NO. 5.

THE RIGHT TO PRIVACY.

"It could be done only on principles of private justice, moral fitness, and public convenience, which, when applied to a new subject, make common law without a precedent; much more when received and approved by usage."

WILLES, J., in Millar v. Taylor, 4 Burr. 2303, 2312.

THAT the individual shall have full protection in person and in property is a principle as old as the common law; but it has been found necessary from time to time to define anew the exact nature and extent of such protection. Political, social, and economic changes entail the recognition of new rights, and the common law, in its eternal youth, grows to meet the demands of society. Thus, in very early times, the law gave a remedy only for physical interference with life and property, for trespasses *vi et armis*. Then the "right to life" served only to protect the subject from battery in its various forms; liberty meant freedom from actual restraint; and the right to property secured to the individual his lands and his cattle. Later, there came a recognition of man's spiritual nature, of his feelings and his intellect. Gradually the scope of these legal rights broadened; and now the right to life has come to mean the right to enjoy life,—the right to be let alone; the right to liberty secures the exercise of extensive civil privileges; and the term "property" has grown to comprise every form of possession—intangible, as well as tangible.

Thus, with the recognition of the legal value of sensations, the protection against actual bodily injury was extended to prohibit mere attempts to do such injury; that is, the putting another in

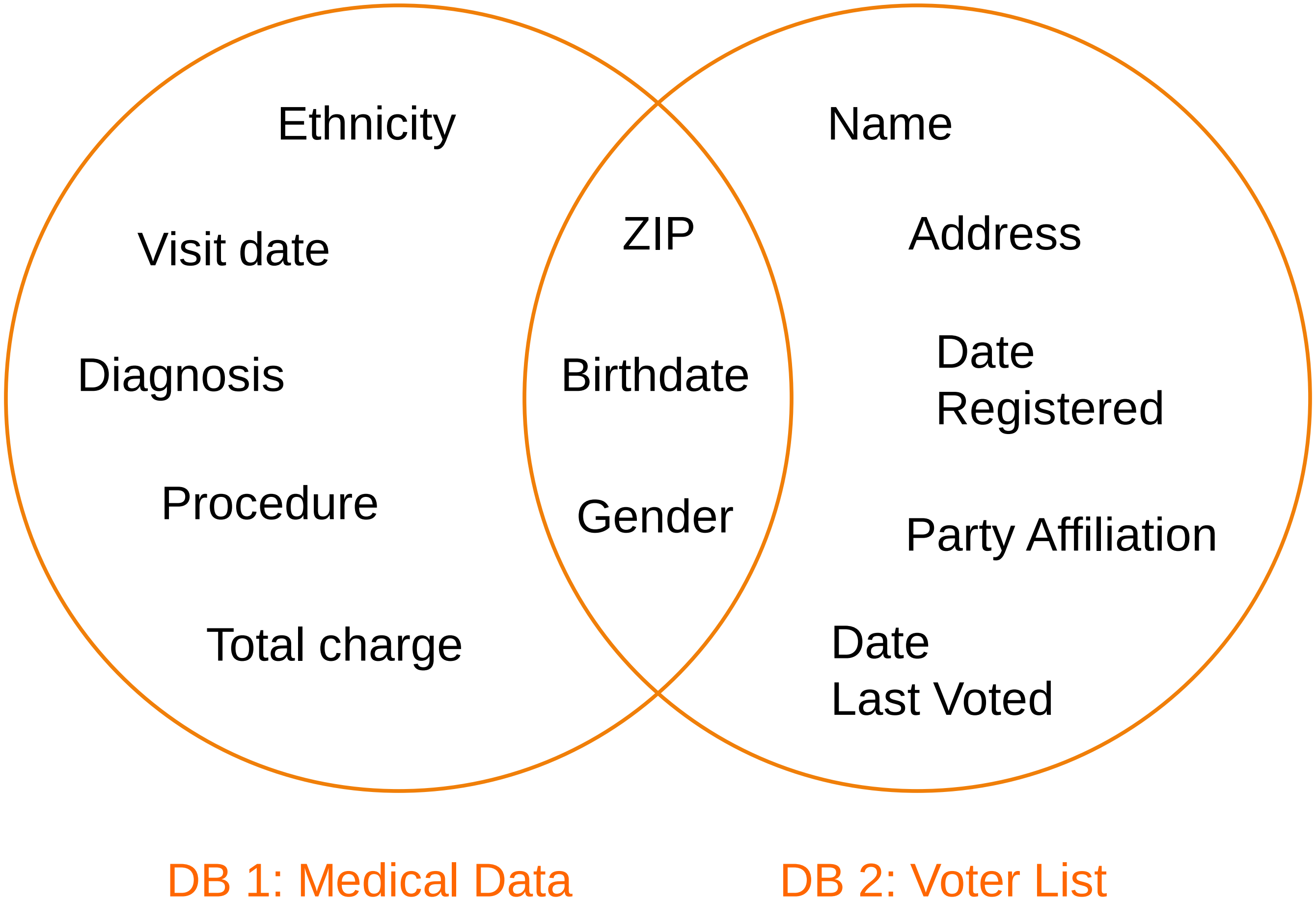
Sensitive Information in Datasets

- Dalenius was one of the first to consider privacy in statistical databases stating that

"Anything that can be learned about a respondent from a statistical database can be learned without access to the database"

- Assume that there exists a national database of average heights of women of different nationalities
- Adversary wants to determine the height of Terry Gross with access to the statistical database on average heights
- Auxiliary information is known that "Terry Gross is two inches shorter than the average Lithuanian woman"
- An adversary can learn Terry Gross's height only if he has access to both pieces of information

Dataset Privacy – Linking Attacks



Dataset Privacy – Famous Attacks

- Netflix dataset released for Crowdsourcing was de-anonymised by joining onto a public IMDB dataset (2006)
- A health dataset from Massachusetts hospital was de-anonymized by joining onto a public voting database (1997)
- AOL public released 650,000 user search queries leading to the using being de-anonymized. AOL faced legal repercussions (2006)
- Genome Wide Association Studies (GWAS) datasets were found reliably useful in identifying participants with certain ailments. Datasets are no longer public.
- MIT discovered that using four spatial-temporal points from a mobility database, 95% of users could be uniquely identified (2013)

k-Anonymity

The principal of k-Anonymity

The principal of k-Anonymity states that the information for each person contained in the release cannot be distinguished from at least k-1 individuals whose information also appears in the release

- Attributes are Quasi-identifiers if they are not unique identifiers but can be combined with other attributes to identify an individual.
- In order to make a dataset k-Anonymous quasi-identifiers need to be generalized or suppressed.

Name	DOB	Gender	Zipcode	Disease
Andre	21/01/1976	Male	53715	Heart Disease
Beth	13/04/1986	Female	53715	Hepatitis
Dan	21/01/1976	Male	53703	Broken Arm
Ellen	13/04/1986	Female	53706	Flu

→

DOB	Gender	Zipcode	Disease
1976	Male	5371*	Heart Disease
1986	Female	5371*	Hepatitis
1976	Male	5370*	Broken Arm
1986	Female	5370*	Flu

Attacks on k-Anonymity

- k-Anonymity while a step in the right direction, does not protect from homogeneity and background knowledge attacks

Bob	
Zipcode	**Age**
47678	27

Homogeneity Attack

Zipcode	Age	Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
4790*	>=40	Flu
4790*	>=40	Heart Disease
4790*	>=40	Cancer
476**	3*	Heart Disease
476**	3*	Cancer
476**	3*	Cancer

A 3-anonymous Patient Table

Carl	
Zipcode	**Age**
47673	36

Background Knowledge Attack

l-Diversity

The principal of l-Diversity

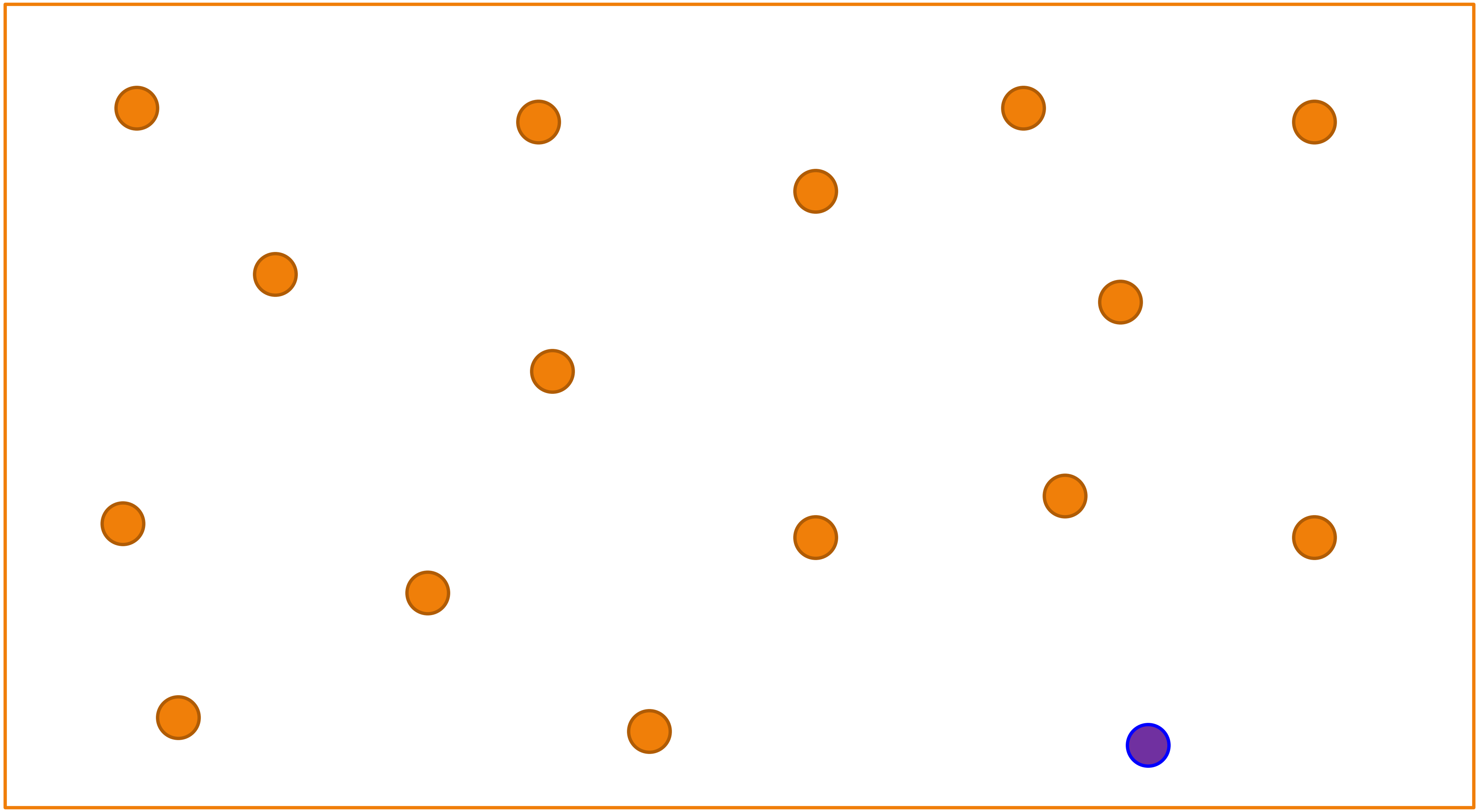
A q-block is l-diverse if contains at least l "well-represented" values for the sensitive attributes S. A table is l-diverse if every q*-block is l-diverse*

Race	Zip	Disease
Caucas	787XX	Flu
Caucas	787XX	Shingles
Caucas	787XX	Acne
Caucas	787XX	Flu
Caucas	787XX	Acne
Caucas	787XX	Flu
Asian/AfrAm	787XX	Flu
Asian/AfrAm	787XX	Flu
Asian/AfrAm	787XX	Acne
Asian/AfrAm	787XX	Shingles
Asian/AfrAm	787XX	Acne
Asian/AfrAm	787XX	Flu

Quasi-identifier equivalence class must have diverse sensitive attributes(s)

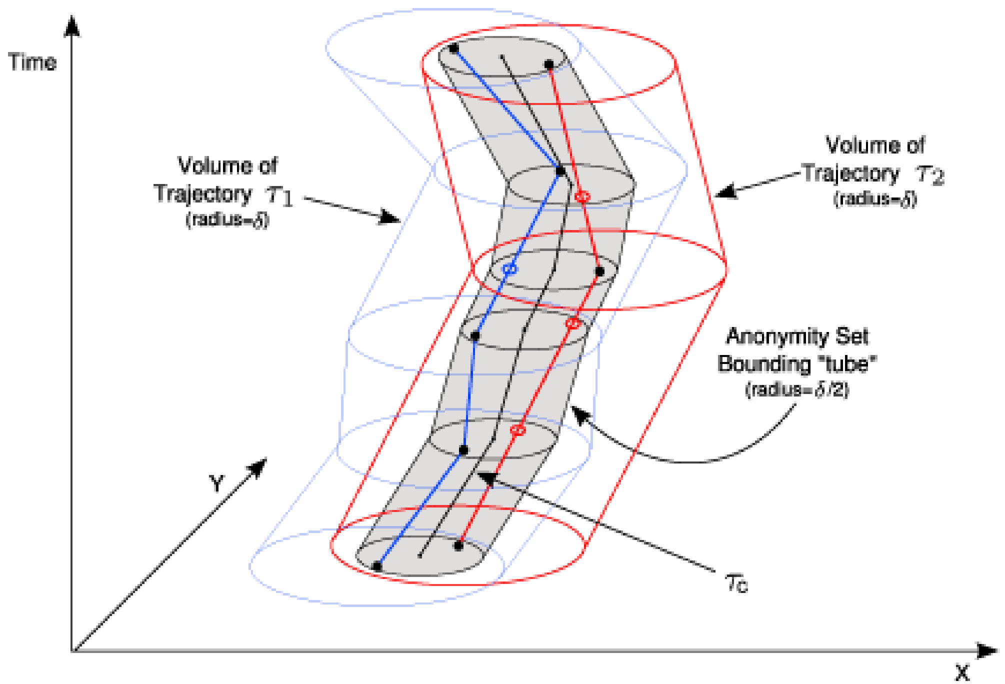
Location Privacy

- Spatial k-Anonymity can be applied to protect a user's location



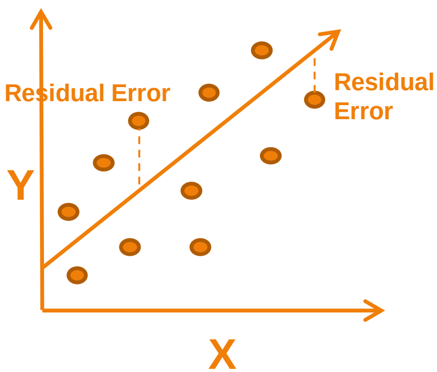
Trajectory Privacy

- In the paper Never Walk Alone, authors make use of impression of GPS coordinates to that a trajectory within a cylinder is k-Anonymous to other trajectories within the cylinder.



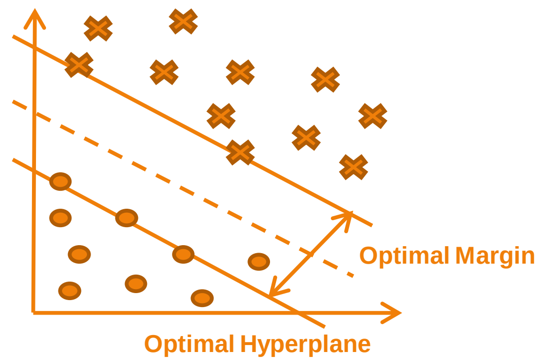
Common Data Mining Techniques

Linear Regression



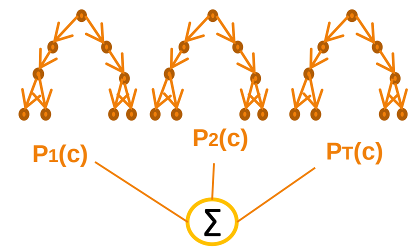
Finds the relationship between two variables by fitting a linear equation

SVM



Machine learning technique based on the principal that you can define an optimal linear decision boundary

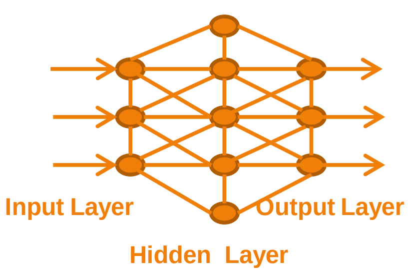
Random Forest



$$P(c|v) = \sum_{t=1}^{T} P_t(c|v)$$

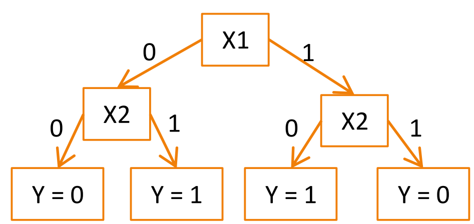
Extending of decision trees is a Random Forest. Creates an ensemble of decision trees.

Neural Network



Neural networks are a supervised machine learning technique. Inspired from how the central nervous system and the brain works in biology.

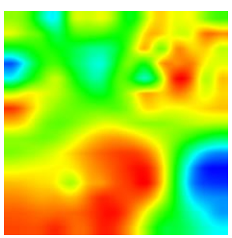
Decision Tree



XOR Function Decision Tree

Builds off the concept of decision trees. Predicts a target variable given a complex series of inputs.

SOM

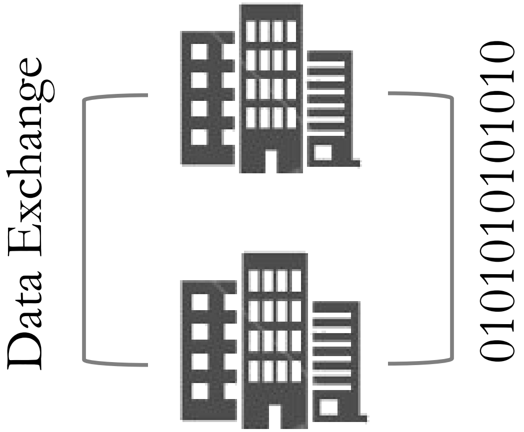


Phone Ringing

Unsupervised modelling technique that produces two dimensional visual representations are utilised to draw inferences from the data.

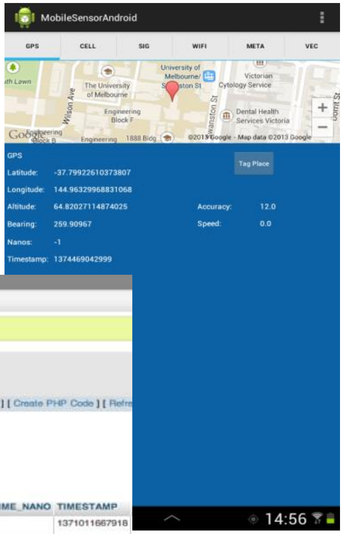
Smartphone Privacy

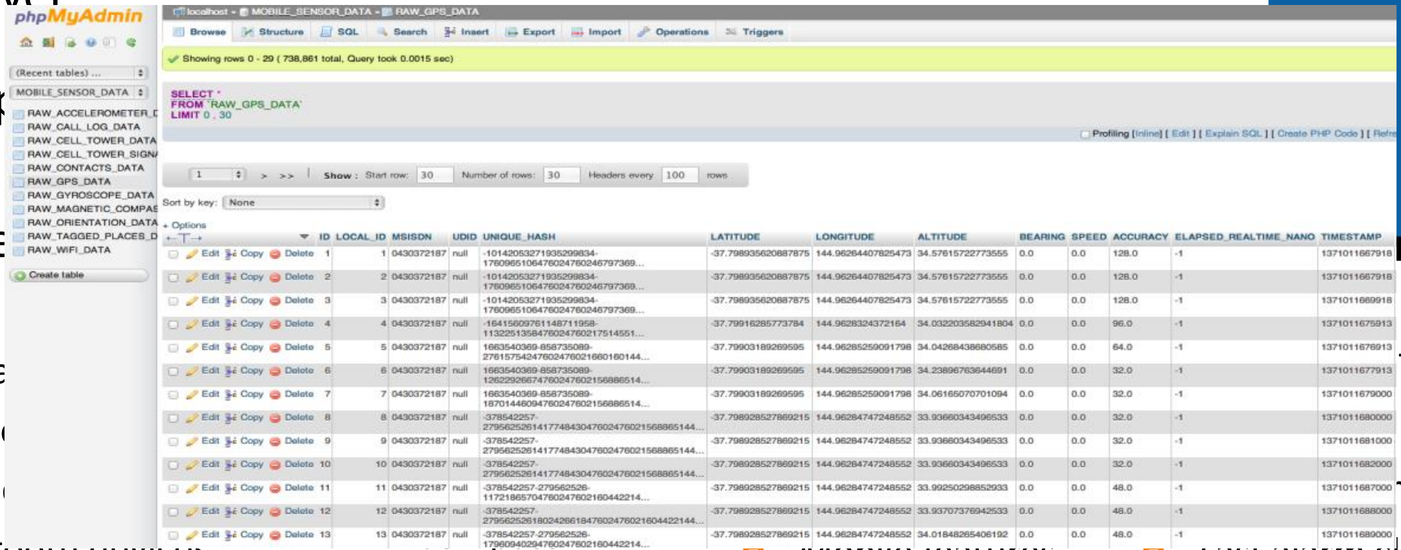
- Sensitive mobile information is accessed via standard developer APIs
- Data is commonly exchanged amongst third parties
- Diagnostic data is commonly sent to developers for debugging purposes
- **We hypothesize that diagnostic mobile data commonly considered to not be sensitive can identify an individual**
- Surveys show user comprehension of privacy is low but users do express concern
- **In practice, with current platforms it is hard for a user to detect current privacy threats apps pose**



Data Capture via Mobile Sensor

- **Android app developed with the intention of capturing all information possible using only the standard API**
- App **runs in the background** and sends data to a remote server
- App
- The

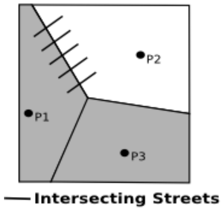


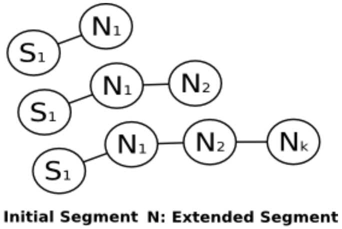


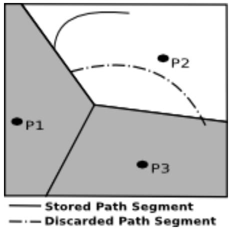
- GPS a
- Cell t
- WIFI
- Bluetooth devices
- Apps Information
- App usage
- True Compass
- Orientation
- Network Traffic
- Mobile features
- File names
- Calendar entries
- f active
- ng preference
- Last alarm clock set
- SMS Messages
- CPU/RAM usage

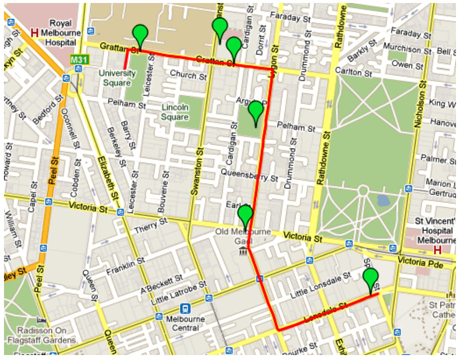
Published in CIKM 2014
Trajectory Inference Attack System

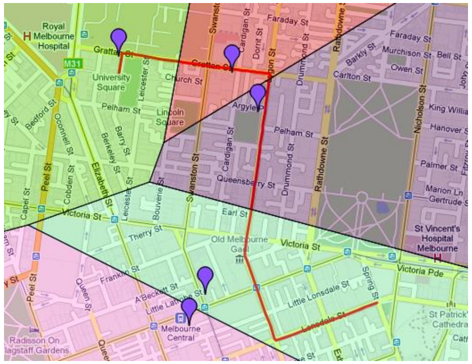
- **Perform a maximum movement attack with the use of a Voronoi diagram for POIs**
- Summarised Algorithm Steps:
  - Obtain Voronoi edge between the first and second points
  - Create paths from intersecting streets by obtaining connected streets and following them (depth-first-search)
  - If expanded path segment becomes longer than maximum speed bound or not in the destination Voronoi cell then discard it
  - Expand set of paths generated until they cross each Voronoi cell.











Trajectory Inference Attack System

- Used 30 modern cloud computers provided by NeCTAR
- Run experiments in a distributed manner
- **Evaluated on 283 real routes in Beijing**

Results:

POI	R = 50	R = 100	R = 250	R = 500
400	27.63	38.9	51.43	64.25
800	34.94	47.73	60.97	73.45
1600	**39.05**	**54.05**	**69.92**	**81.18**
3200	36.12	49.45	64.11	75.12

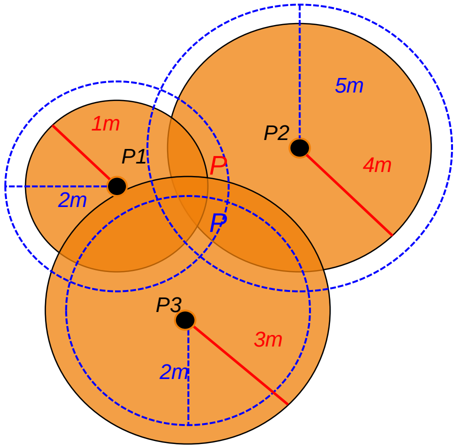
Audible Bluetooth Beacon Data

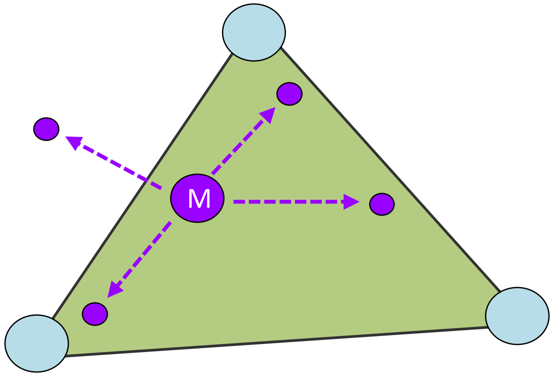
- Over a three week period, we ran a preliminary experiment using only the Bluetooth device discovery feature
- The mobile was left on a desk inside a masters workbook and was not moved.
- Further investigation is needed to determine if a meeting pattern could be mined to determine as opposed to just being in proximity to one another
- **Combining the following should give a good indication of proximity:**
  - Time the sensing device can sense another device
  - Signal strength to approximate distance
  - External social network data to determine if user’s know each other

Audible Bluetooth Beacon Data

Published in Sigspatial 2015
Indoor Localization using Bluetooth

- Indoor localization is very challenging due to signal attenuation and severe multipath propagation
- No indoor positioning technology has reached full mainstream adoption



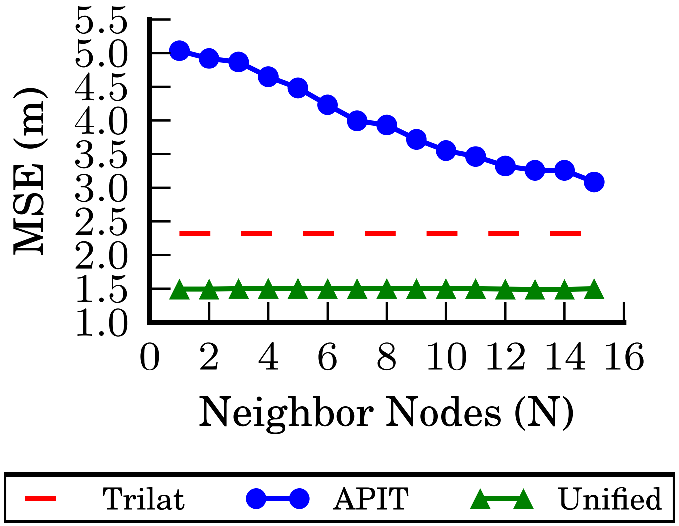


- Range-based methods are susceptible to measurement errors
- Range-free methods are robust, however susceptible to imprecision

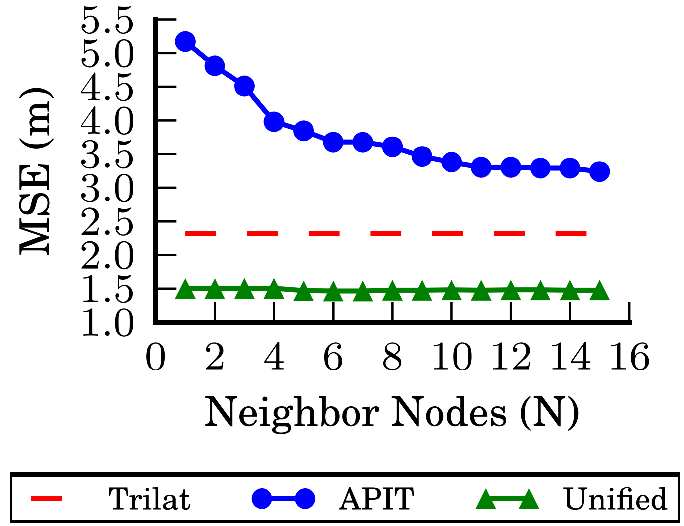
Key Insights from Combining Range-Based and Range-Free

- **Key Observation**
  - Range-based methods do not work well for refinement!
- **Our Method**
  - Combine range-free with range-based localisation
  - First apply a fine-grained method to obtain an initial position
  - Then use coarse-grained localisation for refinement
  - Does not work well in reverse!

Localization Results



Uniform Placement

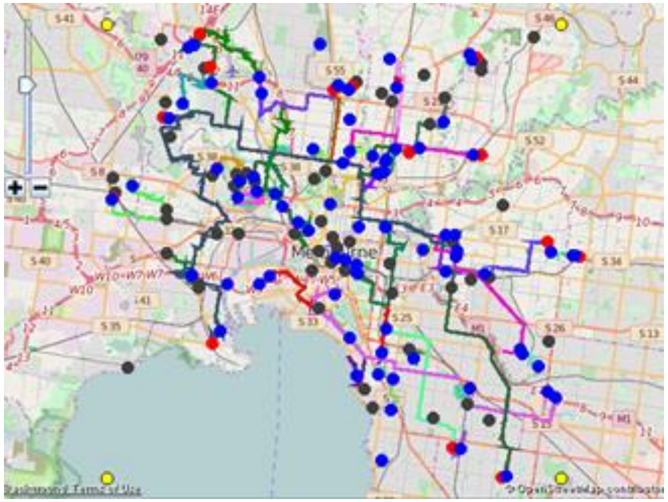


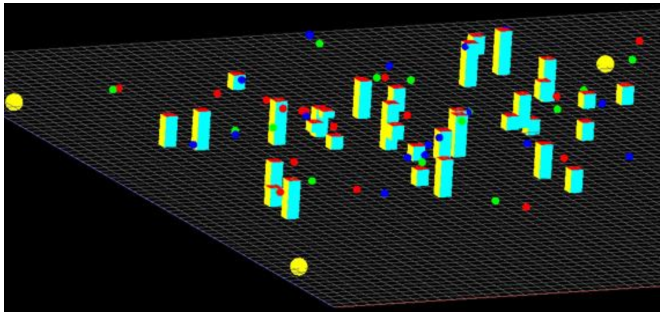
Random Placement

- Unified approach performs better than Trilateration or APIT in isolation
- Only one neighboring node is required to significantly refine a positioning estimate initially positioned from a range-based method
- Technique can achieve accuracies of under 1.5m

Accepted by SIGSPATIAL 2016
Mining City-Wide Encounters in Real Time

- **Key Problem**
  - Smartphones and wearables are capable of sending user locations in near real-time
  - As people travel, they may have encounters with one another
  - **Our aim is to in detect encounter patterns of travelling individuals**
  - Current spatial indexing techniques are not fast enough to capture real-time encounters



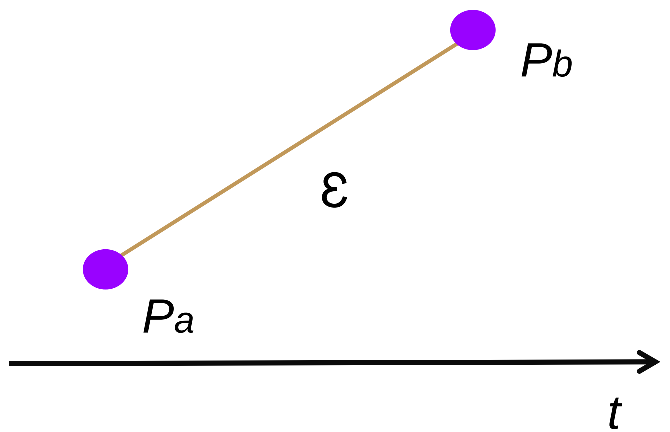


Individuals travelling around the city and a 3D representation of TimeGrid

Constraint Nearest Neighbor Queries (c-NN)

Key Proposal

- A c-NN query only search neighbors that are in proximity and have been in the same area for a certain amount of time
- This can be performed much faster than using classical spatial indexes to run NN queries



Mining City-Wide Encounters in Real Time

- A potential encounter can occur when people are in close proximity
- It is assumed people are within proximity when the distance between them is within ϵ distance
- We define the constraint nearest neighbor (c-NN) query to return objects in proximity at a given a location.
- People that are being monitored for encounters can be represented as a set. Let q be the number of people in the search space and P be a finite set of people.

$$P = \{p_1, p_2, p_3, ..., p_q\}$$

- Let L be the set of locations where a person can be located as well as an encounter may occur.

$$L = \{l_1, l_2, ...\} \text{ where } l_i \in \mathbb{R}^2$$

- Let T be a set of timestamps used to indicate when encounters occur and where a person is located at a particular point in time

$$T = \{t_1, t_2, ..., t_j\} \text{ for } 1 \leq j < \infty$$

Mining City-Wide Encounters in Real Time

- A person $p_a \in P$ is at a location $l_m \in L$ is returned by the function $Loc_t(p_a)$ at a point in time $t \in T$ is returned as follows

$$\mathrm{Loc}_t(p_a) = l_m \text{ where } t \in T,\ p_a \in P,\ l_m \in L$$

- Consider the two locations $l_m \in L$ and $l_n \in L$, they are in ϵ proximity if the following condition is satisfied

$$\mathrm{Dist}(l_m, l_n) \leq \epsilon$$

- Let $\text{c-NN}_t(p_a)$ be the set of people in proximity to p_a defined as

$$\text{c-NN}_t(p_a) = \{p \in P \mid \mathrm{Dist}(\mathrm{Loc}_t(p_a),\ \mathrm{Loc}_t(p)) \leq \epsilon\}$$

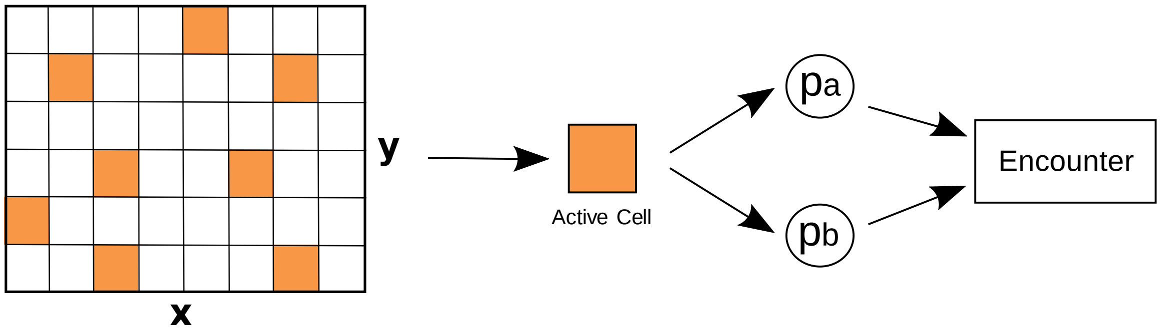
Mining City-Wide Encounters in Real Time

- Let E_t be the set of all people in proximity that have encounters, defined as

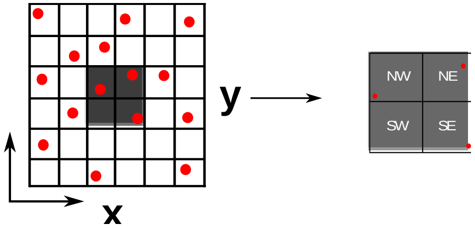
$$E_t = \{\mathcal{P}(\text{c-NN}_t(p)) \mid p \in P\}$$

- In cases where encounters need to be found at a given location, this can be defined as follows

$$E_t(l_i) = \{\mathcal{P}(\text{c-NN}_t(p)) \mid p \in P,\ p \in LocP_t(l_i)\}$$



Searching Active Cells for Encounters



Using QuadCells

Mining City-Wide Encounters in Real Time

- In order to mine for encounters efficiently, we propose the use of proximity and time constraints to significantly reduce the search space
- We propose the use of a spatial index that exploits these properties
- In order to search for people that are within proximity to one another quickly, a grid structure can be constructed and used as a spatial index
- Each person $p_a \in P$ is positioned at a location $l_i \in L$, all locations are within $\mathbb{R}^2$
- The space itself can be partitioned into a grid which can then in turn be used to index each person in the set P
- We define a grid overlaid with each cell to be of size $\epsilon/\sqrt{2}$
- Locations within a grid cell would be within ϵ distance from one another

Mining City-Wide Encounters in Real Time

- The grid over the entire space is defined as G with a side of δ
- The function $\text{CellID}_t(p_a)$ where $p_a \in P$ returns the index of a cell a person is in defined as

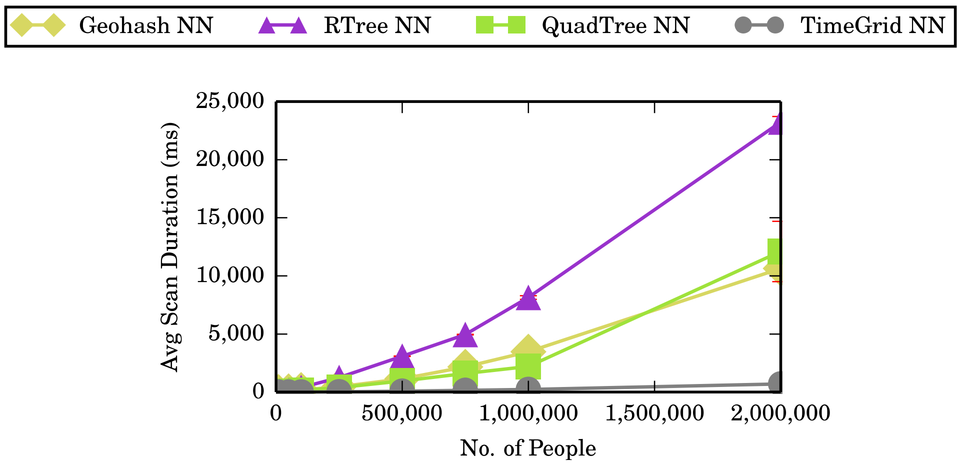
$$\text{CellID}_t(p_a) = \left(\left\lfloor \frac{x}{\delta} \right\rfloor, \left\lfloor \frac{y}{\delta} \right\rfloor\right), \ (x, y) = Loc_t(p_a)$$

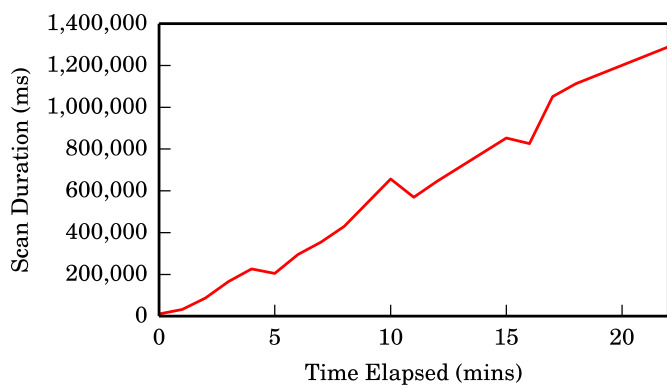
- With these functions defined, $c - \text{NN}_t(p_a)$ where $p_a \in P$ can be defined using TimeGrid as follows

$$\text{c-NN}_t(p_a) = \left\{ \begin{array}{ll} \text{CellP}_t(c) \ \cup \ p_b \ — & \\ & c = \text{CellID}(p_a), \\ & p_b \in \text{CloseQuadsP}_t(c), \\ & \text{Dist}(\text{Loc}(p_a), \text{Loc}(p_b)) \leq \epsilon \end{array} \right\}$$

- Assuming that people are distributed uniformly at random in an area would lead to an average case of $\Theta(m\binom{a}{2})$
- In practice, small values of ϵ, values of a would be small, average case closer to $\Theta(m)$

Results



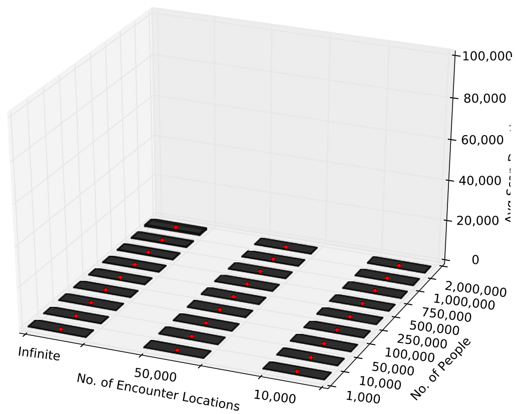


TPR Tree Performance as Time Increases

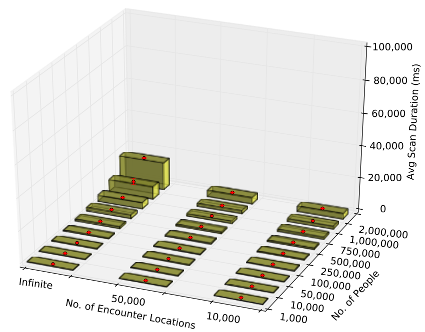
Parameter	Description
p	Number of people in the search space
s	The speed a person is travelling
e	Proximity distance threshold
t	Update frequency of scan for TimeGrid

- Mining encounters using the TimeGrid NN approach outperforms conventional methods!

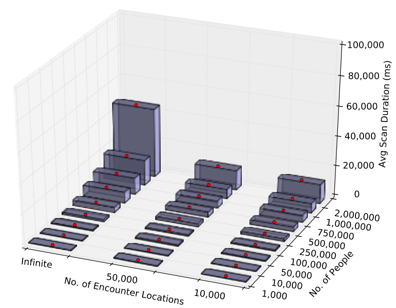
Mining City-Wide Encounters in Real Time



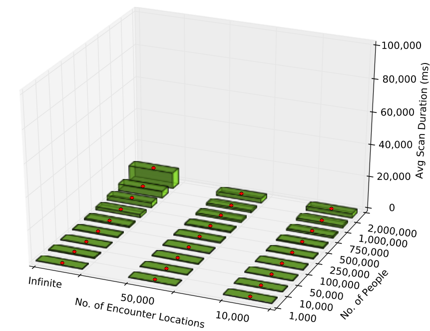
TimeGrid NN - Uniform



Geohash NN - Uniform

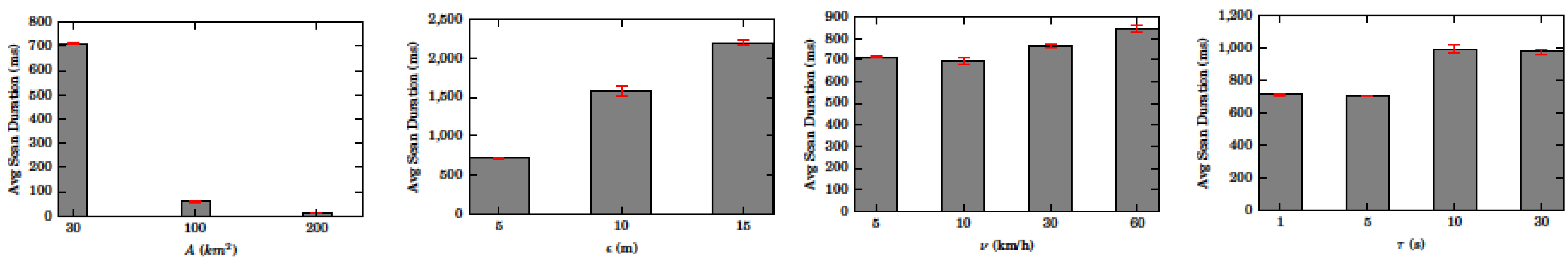


R-Tree NN - Uniform



QuadTree NN - Uniform

- Mining using the TimeGrid approach outperforms conventional methods!



Varying Parameters of the TimeGrid Algorithm - Uniform

Published in MUM 2014
Device Analyser Dataset

- Device analyser is an Android App that provides personal analytics in exchange for anonymized data sourced from the University of Cambridge
- Dataset contains data for ~13,000 Android devices data with ~100 billion entries
- Entire dataset is 7TB and contains very verbose event information
- **We aggregated the data to daily level and tested for correlations between derived features**

Device Analyzer

Device Analyzer Dataset: https://deviceanalyzer.cl.cam.ac.uk/

Methodology

Preprocessing

- Iterate over every data file for each user
- Extract handset ID and date keys for indexing the data structure
- The handset ID and date as a hash to store daily data
- For numerical features store SUM, COUNT, AVG, MIN, MAX
- For categorical features e.g. system settings lock mode, store the value at the end of the day
- Output file for each device
- Combined into a single file and uploaded into relational database

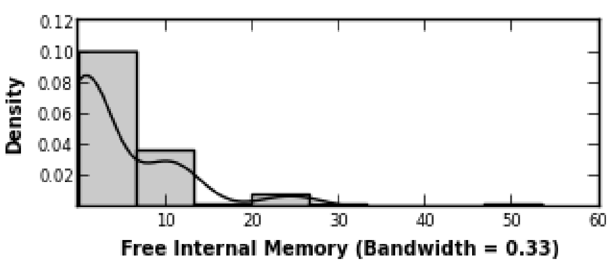
Implementation

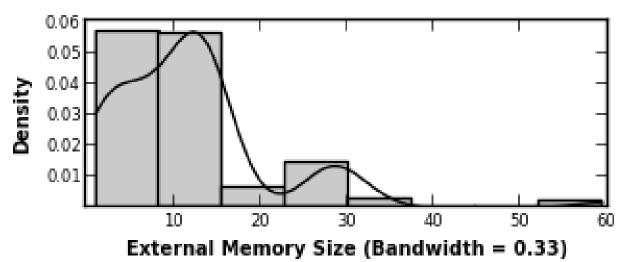
- Use days instead of time periods
- Implemented a parser in C to aggregate data to daily level
- C was chosen over existing DB technologies to aggregate due to the size of the data
- Used a MySQL relational database
- Web app in PHP/MySQL displays selected data
- Kernel Density Estimation of continuous features was performed using the Python scipy stats package
- Tested for correlations and produced models using R

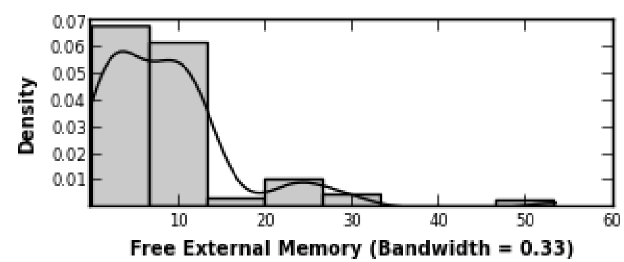
Features Distribution

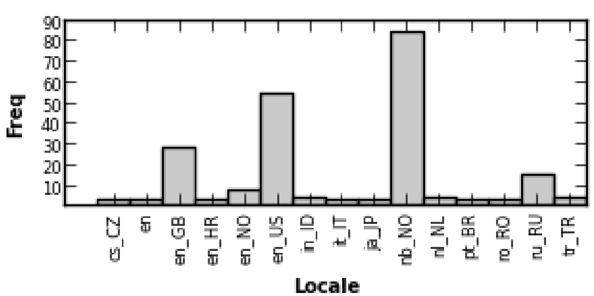
Feature	% Off	% On	% Null
System Settings Lock	52	24	24
System Settings Sound Effects	50	36	14
System Settings Device Stay On	85	9	6

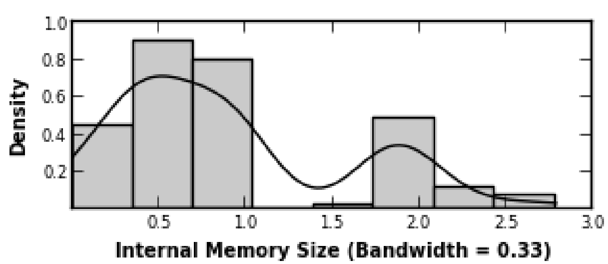
Table 1. System Settings Features Distribution

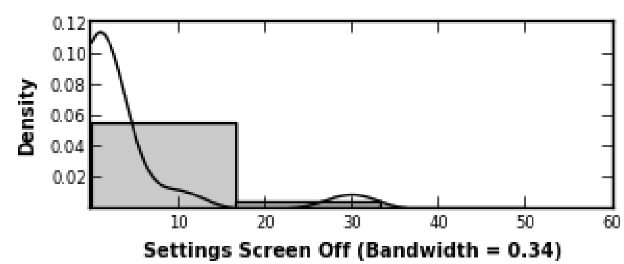












Experiments

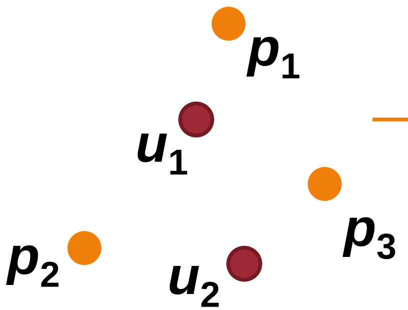
Train/Test Split(%/%)	Accuracy (%)	Macro Avg Precision	Macro Avg Recall
70/30	93.75	0.921	0.949

Table 2. Experimental Results Using a Naïve Bayes Classifer

- Using only the diagnostic features, the model produced by a Naïve Bayes classifier was accurate
- Analysis sample contained 223 days worth of data in which 66 user profiles could be uniquely identified
- Only devices with at least three days of data was analysed
- Numerical features were scaled based on the maximum for the respective feature

Published in Ubicomp 2013
PrivateMeetUp

- **Organize meeting locations using a crowd sourced private and decentralized approach**
- Convert the 2-dimensional (2D) space in to an 1-dimensional (1D) imprecise distance space
- Reveal a range of distances (i.e., bucket) in which the user's actual distance to a POI falls into
- Reduce the degree of imprecision in the distance space iteratively until the group decides on their meeting place



D	p1	p2	p3
u1	3	5	4
u2	8	5	4

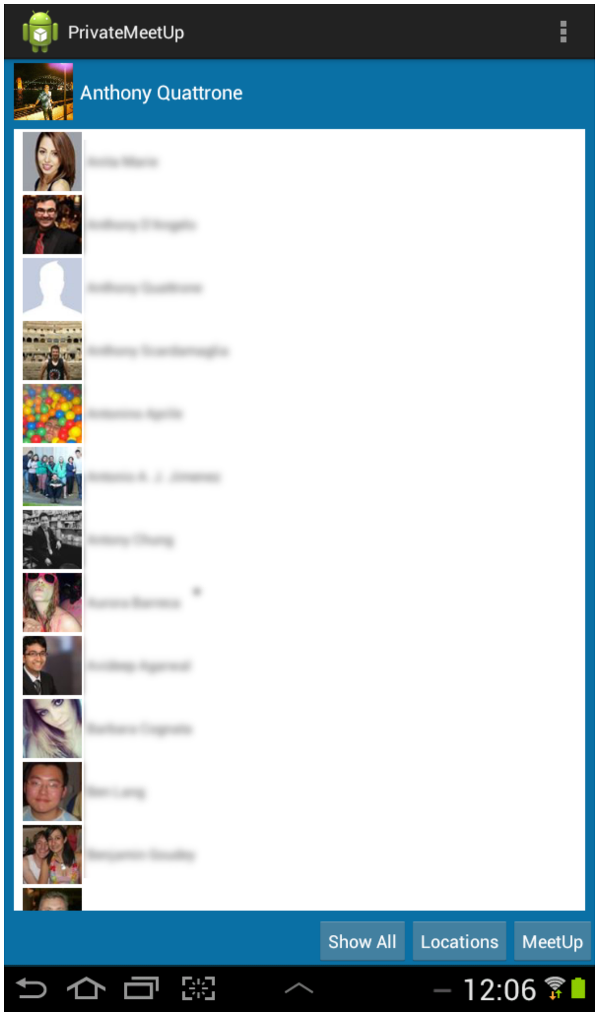
Table 1. Actual distance from users to POI in D

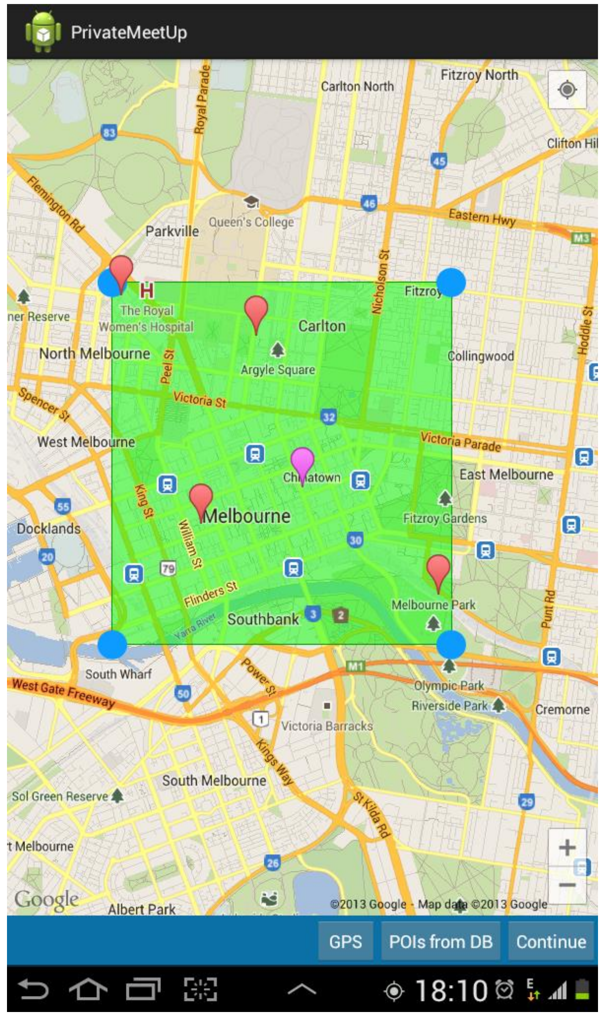
D	p1	p2	p3
u1	1	2	1
u2	2	2	1

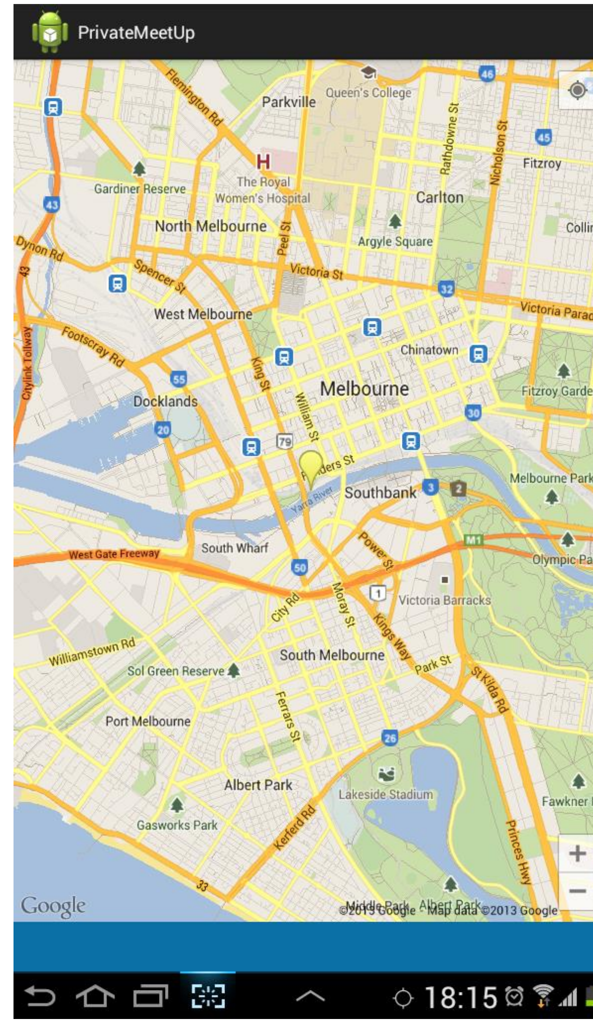
Table 2. Imprecise aggregate distance of POIs

Bucket 1: [0, 4]   Bucket 2: (4, 8]

PrivateMeetUp

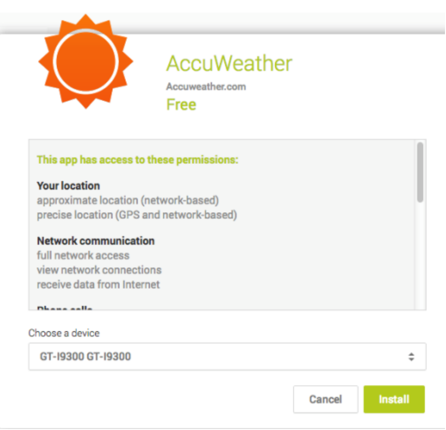






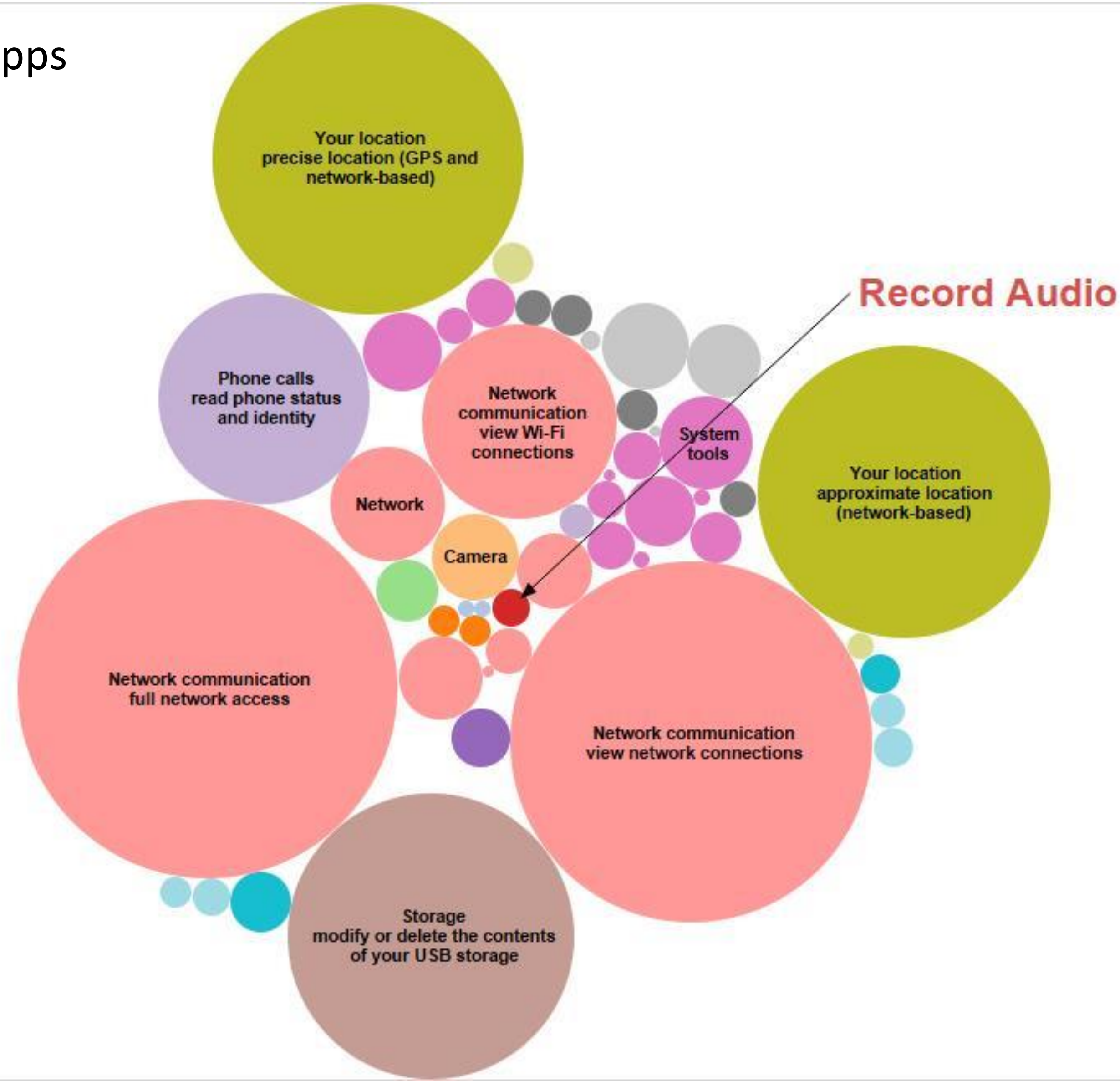
Published in ICICS 2015
Android Smartphones

- Android devices are very popular!
- Android apps are downloaded from Google Play
- **App developers declare permissions the app requires ahead of time**
- **Users are presented with a permission dialog displaying required permissions**
- Permission information of ~17,000 apps was scraped and stored in a database for analysis
- **Results indicate many apps are requesting excessive permissions**



What Data are Your Apps Looking At?

- Weather Apps



Is Android User Privacy Protected?

- **Many users do not easily comprehend the implications of granting third party apps permission to access data**
- No current platform has achieved a good balance between:
  - Control
  - Information
  - Interactivity

- It was found only 15% of the participants paid attention to the permissions at installation time
  - Highlights the need for improved user comprehension

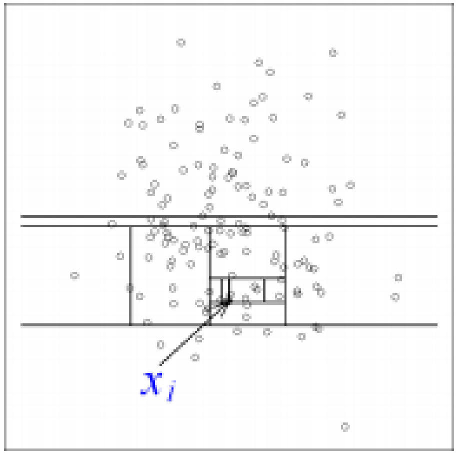
K. W. Y. Au, Y. F. Zhou, Z. Huang, P. Gill, and D. Lie, "Short Paper: A Look at Smartphone Permission Models," in SPSM 2011.

Detecting Privacy Invasive Apps

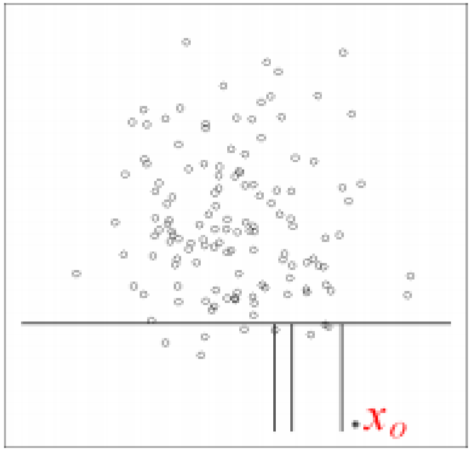
- **Key Issue**
  - How do we distinguish between apps that:
    - Require permissions to improve app functionality
    - Those not following the least privileged path?
- **Key Observation**
  - Compare a target app to apps considered to provide similar functionality
  - Google Play provides a list of similar apps for each app in the catalog
  - **This measure can be used to detect outliers via anomaly detection techniques**

Isolation Forest for Anomaly Detection

- Isolation Forest is a relatively new and unique anomaly detection technique
- Prior methods build a normal profile and isolate those that do not conform
- Instead, Isolation Forest builds a profile that explicitly isolates anomalies
- Random partitions are generated in a given dataset
- The less partitions required to isolate a point, the higher likelihood it is a anomaly
- **Only requires a few data points to detect anomalies**



(a) Isolating x_i



(b) Isolating x_0

F. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation Forest,” in ICDM 2008.

Detecting Apps with Outlier Permissions

1. Generate a table for each target app as follows:

App Name	Perm 1	Perm2	Perm3	Perm4
SimilarApp1	0	1	1	0
SimilarApp2	1	1	1	0
TargetApp	0	1	0	1

2. Construct and train an Isolation Forest using Similar App Vectors
3. Evaluate the Isolation Score of the Target App
4. Assign an alert to each score based on a threshold value **ε**

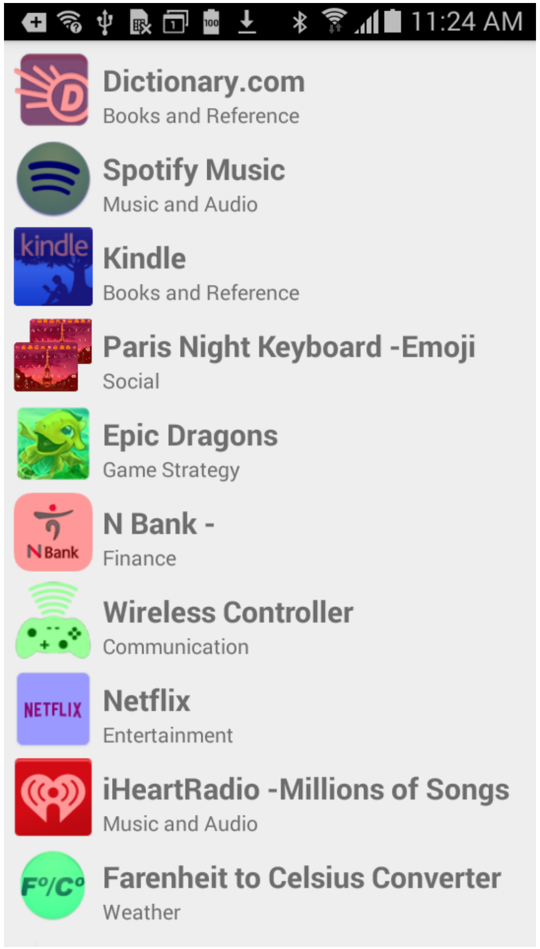
Alert Level	Description
Red	If an IsolationScore is greater than **ε** and uses a sensitive permission
Blue	If an IsolationScore of less than **ε** and uses any sensitive permissions
Green	If an app does not require any sensitive permissions

Category	# Apps	Green (%)	Blue (%)	Red (%)
Communication	381	14.70	73.32	12.07
Social	382	24.35	65.18	10.47
Music Games	320	59.06	30.94	10.00
Action Games	487	32.03	58.52	9.45
Adventure Games	447	39.60	54.36	6.04
Lifestyle	318	42.09	52.53	5.38
Books	356	55.62	39.89	4.49

Number of Outliers Detected Per Category

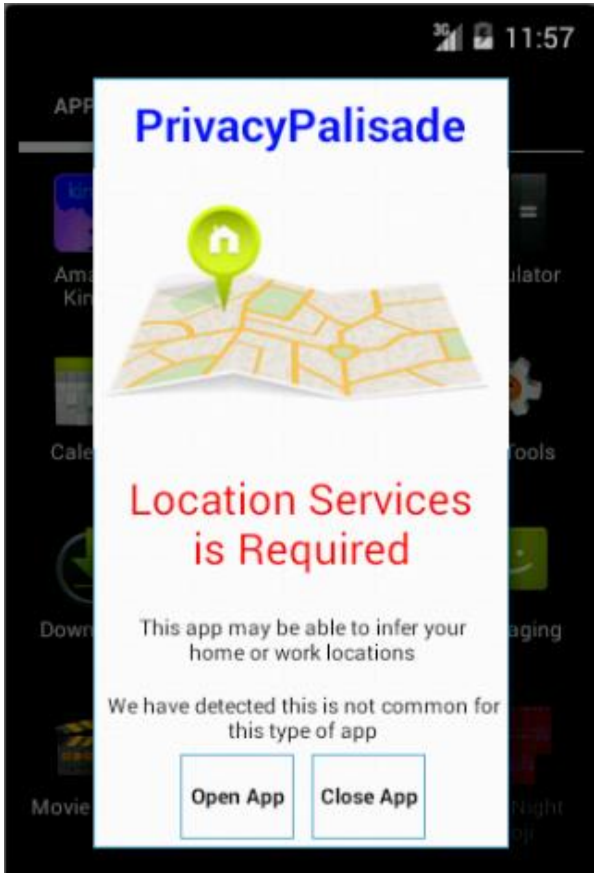
PrivacyPalisade – Android Privacy Protection

- Designed to protect users from potentially privacy invasive apps
- Icons are color coded depending on the invasiveness level
- Users are alerted of sensitive permissions
- Invasiveness is determined by comparing requested permissions to similar apps
- Service runs in the background and retrieves privacy scores from web server
- Integrated into the Android OS Launcher
- Works for newly installed apps and paid apps

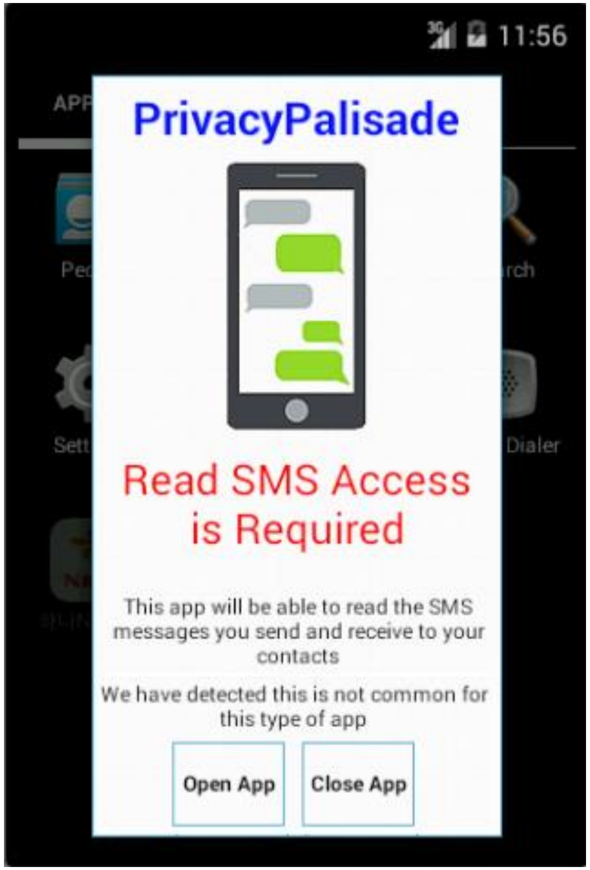


PrivacyPalisade UI

PrivacyPalisade – Alert Dialogs



Location Alerts



Read SMS

Record Audio

PrivacyPalisade – Android OS Modifications

- Android is open source which allows for the creation of custom ROMs
- The native OS Launcher was modified to listen for PrivacyPalisade Broadcasts
- Icons are color coded and uses are alerted of sensitive permissions at launch time
- Android 4.4 KitKat was downloaded from https://source.android.com
- Custom ROM compiled on Ubuntu Linux 14.10

Color Coded Icons

Case Study – iHeartRadio

- iHeartRadio is a popular free music streaming service
- A mobile app is provided for both Android and iOS users
- The app has received 10 to 50 million installs on Google Play
- PrivacyPalisade detected it requested the “Precise Location” permission
- **Precise Location is requested to provide a local radio station search**
- **Approximate Location would suffice**
- Competing apps with a similar number of downloads do not require precise location

Contributions and Future Directions

- Demonstrated how to reconstruct a route using **only POI search results.**
- Proposed a improved indoor localization algorithm that can be applied on Bluetooth to locate users within 1m using **only signal information.**
- Shown how using **only diagnostic data** can be used to train a classifier that can identify people.
- Proposed c-NN queries to perform nearest neighbours queries that only require items in direct proximity fast. Used this **to detect encounters** in real-time.
- Present **PrivacyPalisade**, a system designed to protect user privacy and makes OS level modifications using insights discovered in this research.
- More tools are needed to help users safeguard their privacy. From the results, security software can be better implement to protect user privacy and how to exchange data.
- **We hope to raise user awareness of the potential dangers of certain services and promote stricter privacy and security models.**

List of Publications

Tell Me What You Want and I Will Tell Others Where You Have Been - CIKM '14 Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management.

Is this you?: identifying a mobile user using only diagnostic features - MUM '14 Proceedings of the 13th International Conference on Mobile and Ubiquitous Multimedia

Combining range-based and range-free methods: a unified approach for localization -

GIS '15 Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems

PrivacyPalisade: Evaluating app permissions and building privacy into smartphones - ICICS '15 Proceedings of the 10th International Conference on Information, Communications and Signal Processing

Mining City-Wide Encounters in Real-Time - GIS '16 Proceedings of the 24rd SIGSPATIAL International Conference on Advances in Geographic Information Systems

Protecting privacy for group nearest neighbor queries with crowdsourced data and computing - UbiComp '13 Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computingnces in Geographic Information Systems

On the Effectiveness of Removing Location Information from Trajectory Data for Preserving Location Privacy - IWCTS '16 Proceedings of the 9th ACM SIGSPATIAL International Workshop on

Computational Transportation Science

Data Mining Techniques Comparison

Technique	Linear Regression	Neural Networks	Support Vector Machines	Decision Tree Learning	Random Forest	Self-Organising Maps
Advantages						
Easy to Interpret	✓			✓		✓
Optimal Results for Small Datasets	✓		✓	✓	✓	
Overcomes Noise		✓	✓			✓
Captures Non-Linear Relationships		✓	✓	✓	✓	✓
Captures how Variables are Related to one Another		✓			✓	✓
Scales to Large Data Sets					✓	✓
Disadvantages						
Sensitive to Outliers	✕			✕		
Limited to Numerical Output	✕					✕
Capture only Linear Relationships	✕					
Susceptible from Over-Fitting		✕		✕	✕	
Limited to 2-Class Classification			✕			
Requires Large Datasets to be Accurate		✕				✕