SHIELD+: A Improved Data Verification Framework with Reputation System

Wenqing Yan Panos Papadimitratos
Network Systems Security Group
KTH Royal Institute of Technology
Stockholm, Sweden
(wenqingy, papadim@kth.se)

Abstract -

KEYWORDS

Participatory Sensing Security; Reputation System; Machine learning

I. INTRODUCTION AND BACKGROUND

Recent mobile phones are equipped with a plethora of embedded sensors, and integrate widespread wireless technologies and complex processing capabilities. These technological features have contributed to the emergence of a new paradigm known as participatory sensing (Areputation page1).

In participatory sensing, ordinary citizens collect data from their surrounding environment using their mobile devices and upload them to a campaign administrator using existing communication infrastructure (e.g., 3G service or WiFi access points). (reputation system page 1). After the centralized storage and processing process, end users or other applications can access the results of the sensing tasks made by the application server. For example, the results may be displayed locally on the carriers' mobile phones or accessed by the larger public through web-portals depending on the application needs. (privacy survey page2)

The core idea about PSN is users two-side roles. Except the user character to enjoy the result of centralized processing. They are also expected to contribute and share sensed data from their surrounding environments using their mobile phone. With more consumers, PS system can have much boarder spatial coverage. The broader the participation, the better the results. (Panos page1). However, the inherent openness of PSN (participatory sensing network) makes it easy to collect corrupted data. For instance, users may inadvertently position their devices such that incorrect measurements are recorded, e.g., storing the phone in a bag while being tasked to acquire urban noise information. [reputation system page 1] Besides, malicious users may pollute the sensor data by manipulating for their own benefits.

Threat

Participatory sensing system can be abused both by external and internal adversaries. External attackers are entities without any established connection with the system. They may attempt to gain insights about the participants or the end users. Internal adversaries are active stakeholders of participatory sensing applications, i.e., participants, campaign administrators, and end users. (Privacy in mobile PS)

Without some scheme to process the usability of participants contributed data, the resulting summary statistics will be of little use to the end users. There are two main mechanisms can deal have this function – Reputation system and SHIELD.

Shield

SHIELD is a Verification scheme to classify data sent from a given device based on measurement correlation with other devices to identify malicious nodes polluting the final result. [2017 IoT SENTINEL page7]

Reputation system

Reputation system empowers the campaign administrators to mark reputation scores to contributing devices, evaluating the trustworthiness so that corrupted/malicious contributions are identified. A high reputation scores indicates that a particular device has been reporting highly reliable measurements in the past. Therefore, the administrator can reply more on sensor readings from that device in the future. [reputation system page]

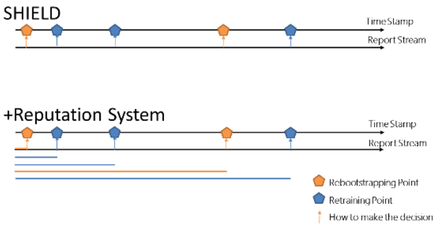
Structure

II. MOTIVATION OF ADDING REPUTATION SYSTEM IN SHIELD

SHIELD operates on the reporting service in participatory sensing system, to assess and remove invalid, faulty data. As a good verification framework, 'the deemed correct data can accurately represent the sensed phenomena, even when 45% of the reports are faulty, intelligently selected by coordinated adversaries and targeted optimally across the system's coverage area.'[panos page 1] Nevertheless, SHIELD owns two short boards. 1) it is agnostic to the cause of faulty reports. Some bad reports may be due to benign impairments of the PS honest clients, but SHIELD does not distinguish deliberately and unintentionally bad reports. 2) If the statistical properties of the sensed phenomenon change, SHIELD need to retrain or bootstrap again to build new classifier model. SHIELD uses predefined threshold to alert the system make the refreshing decision, which only based on single report from different device. [panos page6] This refreshing mechanism is vulnerable, when there are some unstable honest users in the system. In this scenario, the alert may be triggered easily, the system will be trapped in the bootstrapping or training phase. These two phases use efficient clustering method, so the system response capability will be affected.

Reputation system compute device reputation scores based on the device historical information as a reflection of the trustworthiness of the contributed data. [reputation system page1]. Making the refreshing decision based on device report history could enhance the robustness of the SHIELD. Fig.1 shows the main difference of SHIELD and reputation system.

Fig.1 Difference between SHIELD and SHIELD+



III. SYSTEM AND ADVERSARY MODEL

A. SYSTEM

In this paper, we use the basic Participatory Sensing (PS)[trustworthy] system consisting of:

Users: Participants using mobile devices (e.g., smart-phones, smart-vehicles, wearable sensor vests and armbands), equipped with multiple embedded sensors and navigation modules.

Campaign Administrators: Organization, public authorities or individuals [panos page 2 ref 33], initiating data collection campaigns, by recruiting users and distributing sensing tasks to them.

Identity & Credential Management Infrastructure: It is responsible for the Authentication, Authorization and Access Control services by registering users and providing cryptographic credentials. [panos page2 ref 19 22]

Reporting Service (RS): The RS is an access control module that enable registered users to submit data and query the results of a sensing task. SHIELD+ the same as SHIELD operates on the RS. SHIELD+ analyze and remove invalid faulty data with the help of machine learning classifiers and users' trustworthiness scores.

B. ADVERSARY MODEL

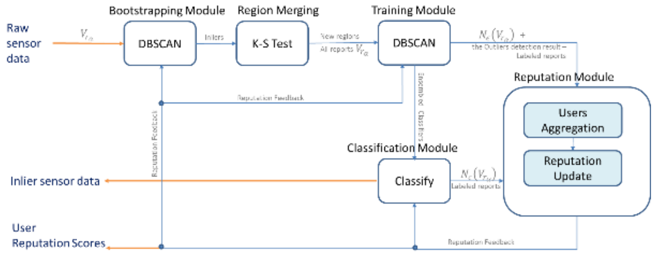
Need to improve based on further research about PSN adversary behavior.

Focus on the internal attack detection.

IV. SHIELD+ FRAMEWORKD

Fig.2 大图 两列

Data flow



A. High-Level Overview

Compared with the original SHIELD, the new SHIELD+ system adds the Reputation Updating Phase to mark reputation scores of contributing devices. Based on the reputation scores, we modify and improve the Classification Phase and the Concept Drift Detection Module.

B. Data Preprocessing

With the help of SHIELD mechanism, Reporting Service (RS) is an agent that discriminate the actual value of the sensed phenomenon, relying on multiple sources of evidence based on the reports. This process is essentially a decision making and a sensor fusion problem. Consistent with SHIELD, SHIELD+ uses *Dempster-Shafer Theory* (DST) to extract the evidence from report.

In this phase, each report is transformed into a probability mass, and computes three metrics a) the hypothesis, H_{max}, with the maximum belief, b) the belief, $Bel(H_{max})$, of this hypothesis, and c) the local conflict of the probability mass, $LCon(m_c)$. These are included in a 3-dimensional feature vector v_{r_a} (one for each report r_a):

$$v_{r_a} = [H_{max}, Bel(H_{max}), LCon(m_c)]$$

m_c denotes the probability mass derived from the user report. [panos]

C. Bootstrapping Phase

For each spatial unit, the system waits until a sufficient number of reports $\{r_a\}$ has been collected and then leverage the DBSCAN (density-based topological clustering) algorithm. The algorithm input is the feature vectors for report in $\{r_a\}$, named $\{v_{r_a}\}$. DBSCAN is a data clustering algorithm with the function of outlier detection. The output of the algorithm is a partition of $\{v_{r_a}\}$ into inliers and outliers. The input of DBSCAN is all the feature vectors $\{v_{r_a}\}$, the maximum distance, ϵ, and the *MinPoints* numbers. The distance function used in SHIELD is Canberra distance metric [panos page 5]:

$$d\left(v_{r_a}, v_{r_\beta}\right) = \sum_{i=1}^{3} \frac{\left|v_{r_a}(i) - v_{r_\beta}(i)\right|}{\left|v_{r_a}(i)\right| + \left|v_{r_\beta}(i)\right|}$$

With the heuristic method introduced in [DBSCAN], we can compute the proper values of ϵ and MinPoints. The ϵ-Neighborhood of a point x, $N_\epsilon(x)$, is defined by:

$$N_\epsilon(x) = \{y \in X : d(y,x) \le \epsilon\}$$

The density of point x, $\rho(x)$, is the number of points in its neighborhood area $N_\epsilon(x)$:

$$\rho(x) = |N_\epsilon(x)|$$

D. Region Merging Phase and Training Phase

Leveraging the inlying reports of each spatial unit, the SHIELD+ then merges neighboring spatial units within which user reports follows almost the same distribution. In this phase, the system recursively calls the two sample *Kolmogorov-Smirnov* (K-S) test. The result is increasing the area of each spatial space and decreasing the number of spatial space.

Once the region merging phase done, the training take place for each formed region. Again, leveraging the DBSCAN clustering algorithm over reports from all the new merged spatial units. The output is a labeling of all user reports (of a region) into inliers and the outliers. Besides, there is one extra important output - the density of each input report, $\{N_\epsilon(v_{r_a})\}$.

E. Reputation Bootstrapping Phase

After the Training Phase, each report has a label（inlier or outlier) and the density value $\rho(x)$. In DBSCAN algorithm the rule to classify the reports is: if $\rho(x)$ is smaller than MinPoints, then this report is regard as outlier. Therefore, the density denotes the importance of each report. This can be the evidence to compute the reputation scores in the Reputation Bootstrapping Phase.

Firstly, for each formed region (Sec.D), the system merges the report according to the contributing user's ID. In the beginning of the bootstrapping (Sec.C), the system need to collect sufficient number of reports. If there are n users in the merged region. Each user may send one or more reports. Based on the user ID, the system classifies the reports set into n subsets. Then, for each subset, (each subset represents a user, $User_i$), compute the average density value, $\overline{\rho_{\iota,0}}$. With this density value, the system computes the initial reputation scores for every user in the Participatory Sensing Network. In order to simplify, we use $\rho_{I,0}$,instead of $\overline{\rho_{\iota,0}}$ in the following paper.

In the reputation system. We tend to gradually build up trust in another user after several instances of trustworthy behavior. However, we rapidly tear down the reputation for this individual if we experience dishonest behavior on their part even in a handful of occasions. [reputation system page 5] According to the reputation behavior, SHIELD+ use the *Gompertz function* for computing reputation scores.

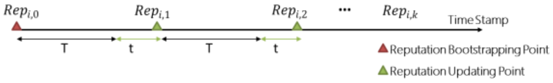
For user i :

$$R_{i.k}(\rho_{i,k}) = ae^{be^{c\rho_{i,k}}}$$

Where a is the upper asymptote; b controls the displacement along the x axis; c adjusts the growth rate of the function. These are the parameter of Gompertz function, and the input is the density value $\rho_{i,k}$. In the Reputation Bootstrapping Phase, the $\rho_{i,k}$ is $\rho_{i,0}$.

F. Reputation Updating Phase

The reputation scores need to be refreshed. Since, the density computing process is resource-intensive. SHIELD+ system set the time interval value T to refresh the reputation scores, $Rep_{i,k}$.



After the T time, the system collects reports in the upcoming t time, $\{r_a\}$, and calculates the density of each report, forming the density set, $\{\rho_{i.k}\}$. k is an integer, representing the refreshing times. The Reputation Bootstrapping process is the 0th (k = 0), then every updating process k plus one. We use the definition epoch k to represent the kth time interval t.

In this phase, each epoch, the system gets a new $\rho_{i.k}$ for each user i. Then, the system transforms $\rho_{i.k}$ to Reputation Scores, $R_{i.k}$, with *Gompertz function*. The input of the function needs to reflect the fact that reputation is the result of aggregating historical device information (i.e., $\rho_{i.k'}, k' = 1,2 \ldots k$). SHIELD implement the aggregating process in [reputation system page 5].

For user i:

$$\rho_{i,k}^{norm} = \frac{2(\rho_{i.k} - min\{\rho_{i.k}\}_{i=1}^{n})}{max\{\rho_{i.k}\}_{i=1}^{n} - min\{\rho_{i.k}\}_{i=1}^{n}} - 1$$

Where $max\{\rho_{i.k}\}_{i=1}^{n}$ and $min\{\rho_{i.k}\}_{i=1}^{n}$ represent the maximum and minimum density values from the density set in epoch k, respectively. Then the input of *Gompertz function* can be expressed as follows:

$$\rho'_{i,k} = \sum_{j=1}^{k} \lambda^{(k-j)} \rho_{i,j}^{norm}$$

Where the summation is used to facilitate the aggretation of historical information while the exponential term, $\lambda^{(k-j)}$ with $0 < \lambda \le 1$, reduce the impact of the past data.

G. Classification Phase

In the bootstrapping and Training Phase (Sec.C and D), the SHIELD system uses clustering unsupervised machine learning method to classify the inliers and outliers. As the high resource consumption ability of clustering, once we get the training result (Sec.D) for the merged spatial region, then the system changes to use classification method to find the outliers. With the input of the labeled reports, the system trains an ensemble of classifiers comprising a random forest, a naïve Bayes classifier and a nearest neighbor classifier. [panos] Then, leveraging the ensemble classifier, the new raw data from the users can directly be grouped into inliers and outliers.

In SHIELD+, the new system keeps all the original processes in Classification Phase, but add one preprocessing step for the new raw data (the input of ensemble classifier).

This preprocessing step is called Reputation Fliter. On the basis of the reputation scores variance between outlier and inliers, the system set a Reputation Threshold, RS . Through the Reputation Fliter, the system removes the reports from the users, which reputation scores $Rep_{i,k}$ is lower than RS, where $Rep_{i,k}$ is the latest reputation scores.

Reputation Fliter, can relieve the load of the classifier and help the system to identify the malicious users and protect the final results from being polluted.

H. Concept Drift Detection Module

Consistent with SHIELD, the model is resposible for detecting the changes in the statistical properties of the sensed phenonmennon. However, instead of keeping monitors the diagreement between the probability mass, SHIELD+ uses Reputation scores to make the decision.