Verification of User-reported Context Claims with Context Correlation Model

Jindan Zhu,[1] Anjan Goswami,[1] Kyu-Han Kim,[2] and Prasant Mohapatra[1]

[1]Department of Computer Science,
University of California at Davis, Davis, CA 95616
{jdzhu, agoswami, pmohapatra}@ucdavis.edu

[2]Hewlett-Packard Laboratories,
Palo Alto, CA 94304
{kyu-han.kim}@hp.com

***Abstract*—Context-aware services nowadays offer incentive to user-reported context information , which inevitably solicits malicious users to cheat by submitting fabricated context claims. Conventional countermeasures based on Trusted Computing Base typically focus on particular context of interest, while disregarding the availability of various types of context information and the intrinsic correlation among them. In this work we propose a context claim verification scheme that interrogates correlated contexts of multiple dimensions to corroborate or contradict the reported context. Specifically, it first learns and models the context correlation with a Bayesian Multinet. Given a claim consisting of reported context and witnessing evidence, the scheme performs Bayesian inference with the evidence to verify the reported context. The verification process is light-weight, and can be applied to arbitrary types of context with a single model learnt. Evaluations on Reality Mining dataset and synthetic dataset validates choice of Multinet for data modeling, and demonstrate the feasibility of our scheme in context verification.**

I. Introduction

The prevalence of smartphone and the advanced sensing capability embedded in it have brought about a new paradigm of mobile services: the context-aware services. Based on the sensed and inferred contexts by user device, service provider can now offer services customized to the changing environment in which the user is present. An emerging business model is readily built on top of the context awareness, as tailored service or incentive is rewarded to users who meet a specific context requirement. A well-known example would be the "check-in" feature of many popular location-based services [1], via which discounts are rewarded to frequent customers of a venue who report the location context. "Pay-as-you-drive" insurance [2] offers clients a lower premium based on their location context and driving pattern. Fitness application [3] uses activity context to monitor workout routine progress, and allows a user to compete with friends accordingly as an incentive to promote persistence.

All of these services rely on context information inferred from sensory data and reported by the user. Therefore the quality of context in terms of both accuracy and authenticity is crucial to context-aware services. Numerous effort has been invested in improving the inference accuracy for various contexts, while protecting *context authenticity* has just begun gaining attention following the emerging applications aforementioned [4]. The existence of incentive provides malicious users with a strong motivation to fabricate their context claims. Such a malicious behavior is not difficult technically on popular mobile platforms, especially when the user has full control of the device [4], [5]. These emerging attacks pose serious threat to context-aware services. To combat the exploits, one approach [6] is to attest the claimed context information inside a trusted environment on user device. This approach makes use of the trusted platform module (TPM) of user device to produce trustworthy digitally signed statements about reported context. Depending on the service requirement, such attestation may happen at every stage of context processing, from raw sensory data to inferred context. Moreover different types of context data typically requires different attestation handling during its processing. Considering the amount of contexts at multiple levels required by various context-aware services, attesting all of them will be a daunting task for developer, which naturally imposes trade-off between ensuring authenticity and overhead in delay, computation, and development [7]. Several application-specific schemes are proposed to protect authenticity of particular contexts, such as in location-based service [8]. However it is difficult to generalize these schemes to protect other categories of context and service. Aforementioned limitations motivate us to look for an alternative scheme for context authenticity verification, which may work as online fraud detection system that flags suspicious claims before engaging intense attestation process. The design should be light-weight and delay tolerant, as well as universally compatible with wide range of contexts and services.

The idea is to make full use of the abundant contexts available and the correlation among them in jointly describing present environment. *Context* in this case does not have a strict definition, and may refer to any piece of information observed or inferred at the moment. We adopt the notion of *situation* proposed in [9] to denote a pattern about the temporal state of user and surrounding environment. Observed contexts are correlated not simply because of their inherent relations of physical nature, but also because of the common effect from varying situation. In a typical situation if one context observation is artificially bent, other observations can contradict the artificial change since majority of the correlation model is intact. Based on this idea, a scheme can attest arbitrary context observation with a pack of correlated contexts, and verify if the observation is valid. The context that a malicious user targets to tamper is denoted as *primary context*, while the set of other available contexts useful for attestation is named *auxiliary contexts*. When attestation with TPM for primary context is not available or too expensive, the scheme can alternatively look at a set of correlated auxiliary contexts that are partly attested or easily attested, and use auxiliary contexts to attest the primary context based on their inherent correlation.

One simple example is illustrated in Figure 1. In this example, a user has only two situations: "Coffee" and "Work"

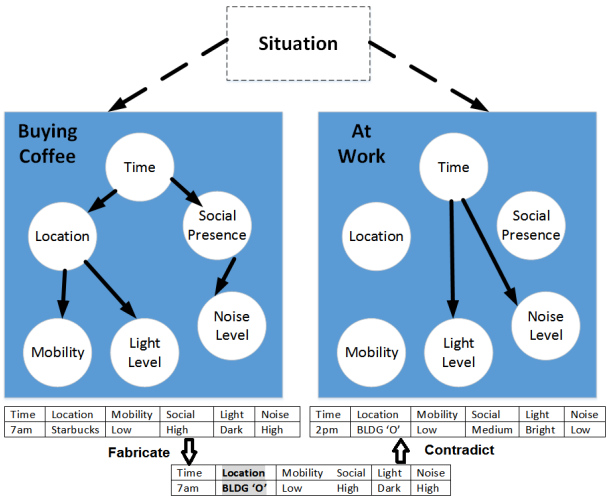


Fig. 1. Correlated Contexts in Describing Situation

which can be captured by six observed contexts: {Time, Location, Mobility Level, Social Presence, Light Level, Noise Level}. For each of the situations, distinctive relation among correlated contexts is formed to reflect the property of situation. The correlation can be visualized with a graphical model, and the table below in the figure simplifies the distributions encoded in the graphical model by showing the prominent case of strong correlation. Suppose a malicious user who is at the **Starbucks** in situation "Coffee" intends to lie about his location context by submitting a fake location **BLDG 'O'**. By reviewing auxiliary contexts captured along with the fake primary context, the scheme can flag a potential fraud since neither situations will produce a high posterior probability given the pack of contexts.

Context correlation has been successfully extracted from user context history in previous studies [10], [11]. The discovered relations are more evident and accurate if for recurring situations, which are usually more appealing to malicious users as primary targets. Proliferation of sensory modules and inference algorithms, as well as technologies exploiting shared context cache [12], ensure the availability of rich user context history of various context dimensions. Despite the fulfillment of these technical prerequisites, there are tremendous challenges in transforming the seemingly intuitive idea into a practical scheme. One of the fundamental questions is: What is the appropriate model for accurately representing the context correlations in diverse situations? Bayesian Networks is a well-studied technique for modeling causal relations and dependency among attributes, with several advantages over other modeling technique such as association rules [11]. It provides sophisticated learning algorithms for discovering patterns from user data, as well as lightweight inference methods based on Bayes rules. Its ability to handle missing values allows us to model various contexts with great flexibility. Nonetheless, generic Bayesian Network is purposed to work with homogeneous dataset so that all patterns can be represented by single model. Everyday user is experiencing numerous situations, in each of which diverse behavior patterns are captured through unbalanced context observation instances. Moreover evolved and transient situations may emerge every now and then. The imbalance and conflicts among situations often cause less obvious pattern obscured and missed by the model learning. In the previous example suppose single Bayesian Network is learned for both situations. Inference about minor situation "Coffee" may incur large error since the model is learnt based on dominating situation "Work".

In this paper, we propose a novel scheme for attestation and verification of user submitted context claims based on a Bayesian Multinet model of context correlation in the presence of conflicting situations. In a training phase the scheme automatically discovers relevant situations from vast user context history, and then iteratively learns Multinet components for individual situations using an Expectation-Maximization (EM) variant. Given a context claim consisting of potentially tampered primary context and attesting auxiliary contexts, the scheme produces posterior probability score for the claim by performing Bayesian inference with the thus-learned model, and finally verifies the authenticity of the claimed primary context. Using our scheme, context claim about recurring situations can be verified efficiently, without resorting to TPM all the time. We justify the adaptation of Multinet model by conducting a comparative study between Multinet (MN) and generic Bayesian Networks (BN) model with real user data as well as synthetic dataset. It shows that MN contributes to large improvement when conflicting situations exist. Evaluation on overall performance is performed on Reality Mining dataset, and the results demonstrate the effectiveness of our scheme in context attestation and verification.

The rest of the paper is organized as follows. In section II we describe the system and threat model used in our scheme, and a brief introduction of graphical models. Section III presents design consideration and analysis, as well as the implementation details of major components of the scheme. Results of system evaluation are given in section IV. Section V reviews existing solutions for context attestation and context correlation modeling. Finally we discuss future work and conclude this paper in section VI.

II. System and Threat Model

A. System Model

We consider a system model of typical context-aware service that involves two parties: *user device* and *service provider*, as illustrated in Figure 2. A user device is equipped with context sensing and inference capability in different domains. For the sake of simplicity we only consider high level contexts (e.g. places instead of coordinates) produced by arbitrary inference methods with acceptable accuracy in the absence of attack. When requesting service, our system on device side will construct and attest context claims before transmitting. From time to time it will also collect context records irrelevant to service request as training data, compiled as context history and sent to service provider. The collection is performed randomly to minimize the possibility of malicious pollution. Although optional, context records attested by TPM can be included to enhance the quality of training data. Contrary to a traditional service provider who is interested only in specific primary context, we abstract the service provider into a generic entity who can acknowledge arbitrary primary context. Apart from providing context-aware service, this service provider is also responsible for storing user context history data, and from which learning the context correlation model. Based on the model, the verification module verifies the authenticity of primary context through Bayesian inference.

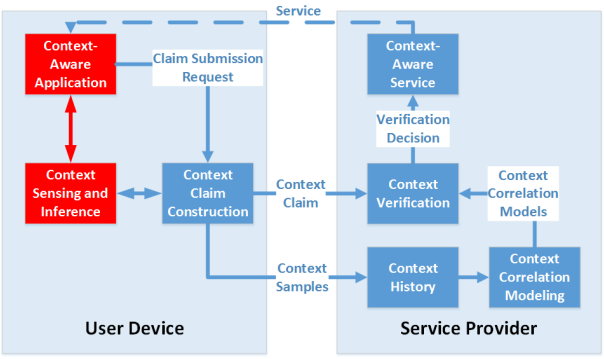


Fig. 2. Context Attestation and Verification Model

Note that in this work, we mainly consider personalized context model for individual user. Different users may possess very diverse habitual pattern that leads to unique context model to each user. A generic model learnt from multiple users can indeed capture certain common patterns among them. However it is not helpful in learning pattern with adequate granularity for particular user especially when the user deviates from general populace. Therefore use of generic multi-user model is discouraged and applies only when personalized model is unavailable due to lack of training data.

B. Threat Model

Our primary adversary is a malicious user whose objective is to fabricate the primary context in a context claim. Such attack may be realized by generating fake sensor readings and manipulating context inference algorithms provided by OS, which is trivial for the capable adversary. However we assume adversary cannot easily breach the sandbox mechanism, and compromise the functionality of our scheme. TPM is still in place, so as a number of TPM attestation techniques applicable to specific contexts. We also assume the OS manages contexts with a context information cache. When a context is sensed, inferred, or attested, it is put into the cache so that other services or applications, including our scheme, can access it without the need of reproducing it. In addition we do not consider physical emulation attack that creates a physical environment emulating the actual environment

Within our attestation framework, the capability of adversary is defined by his knowledge about context correlation model. In this work we begin with the most commonly seen *parochial adversary*, who cheats under impulse and attacks only primary context at a time. Such an adversary operates without the global knowledge about the correlation between primary and auxiliary contexts, nor the attempt to carry out liaised attack on them to improve the credibility of falsified context claim. Auxiliary contexts are assumed either attested thus genuine, or missing from the cache. For instance when cheating a location based service, the adversary may focus on attacking location context by faking GPS reading or interfering with localization algorithm. Nonetheless it is beyond his knowledge about other auxiliary contexts and ability to initiate simultaneous attacks on them. Since auxiliary contexts are collected and assembled into context claim by our scheme, it rules out naive replay attacks using previous context claims. We will leave the case of more powerful adversary to discussion and future work.

C. Graphical Models

Bayesian networks (BN) [13] is a compact graphical representation that has been widely used for encoding uncertainty and dependency. Its abilities to handle missing values and to make efficient inference following Bayesian rules make it attractive in modeling context correlation, as shown in one of its recent applications [11]. One can assign a variable X_i to every context, with set of values expressing all possible observations. Correlation among contexts is naturally visualized through the structure of learnt model, with parameters reflecting conditional probabilities associated each context. Efficient learning algorithms have been proposed [14], [15], in which the dependency between variables are tested according to certain scoring functions such as mutual information.

However generic Bayesian Network learns a single network to model all dependencies among variables, thus cannot handle well the case of *asymmetric independence*[16], where variables are independent for some but not necessary for all of their values. Asymmetric independence corresponds to many realistic scenarios, especially the user context history considered in our case. The dataset consists of context observations from a mix of various situations, each of which may exhibit rather distinct correlation pattern. The problem becomes acute when two types of conflicting situations embodied by unbalanced samples in dataset, a case commonly observed when a new behavior pattern is developing or a seasonal habit coexists with majority norm. Modeling with generic Bayesian Networks usually causes correlation pattern from prominent situation to overshadow the minor ones, thus increasing error when verifying a claim of minor situation.

To address this deficiency, *Bayesian Multinets* (MN) extends Bayesian Networks by creating a mixture model, of which each component is a distinct Bayesian Network corresponding to one kind of dependency relation. A *hypothesis variable*, H, is introduced to represent the possible asymmetric independence relations. Any instance of a given dataset is sampled from one component BN associated with a state of H. By adopting Multinets model, the learning procedure will try first to identify and partition distinctive situations mingled in the dataset, prominent as well as minor ones, by determining the states of variable H. Then for each identified situation and respective data subset, dedicated BN component is learnt to model correlation specific to the situation. By choosing the correct state of H and the corresponding component BN, claim of minor situation can still be accurately verified without the interference from dominating situations. As a result, we choose Muiltinets to model the context correlation, and demonstrate its effectiveness in following sections.

III. System Design

In this section we describe the design details of our scheme, including training and testing dataset studied, the learning algorithm for Multinet model, and inference algorithm for context claim verification.

A. Context History Processing and Claim Construction

User *Context History* is an exchangeable data set represented by a matrix. The columns in the matrix correspond to all observable contexts for user. The rows are concurrently collected context observations of an instance, suggesting a sampling from certain situation distribution. When new context dimension is activated for observation a corresponding column

can be appended to the matrix. If a context is not observed at the time when the instance is collected, its observation is marked "missing" ("[]"). An example of the data representation is shown in Table I, in which the context observations are discretized nominal values which will be explained later as we describe the dataset. *Context Claim* (Table II) has a main body similar to an instance in context history, with additional metadata indicating the primary context. Other observed contexts enclosed are treated as auxiliary contexts used for attesting the primary context.

TABLE I
DATA REPRESENTATION

Time	Loc	Mob	Soc	App	Comm	Batt
11	1	1	1	1	1	2
13	3	2	1	1	[]	1
...	...	...	...	...	...	...

TABLE II
CONTEXT CLAIM: BOXED CONTEXT IS PRIMARY CONTEXT

Time	Loc	Mob	Soc	App	Comm	Batt
11	1	1	1	[]	1	2

1) Reality Mining Dataset: One dataset we used for evaluation is the well-known Reality Mining Dataset [17], which collects smartphone sensory data from 106 participants from MIT. The smartphone records important measurements such as timestamp, associated cellular ID, nearby Bluetooth devices, call logs, battery status, etc. We screen the data by dismissing users with less than 720 hours of data, and those do not show enough location diversity, e.g. large trunk of "Elsewhere", "No Signal" locations. It leaves us a dataset of 26 users, and from which we will be able to infer seven most relevant contexts as follows:

- **Time of the day**: We discretize all timestamps into 24-hour intervals, and process aggregated data of other measurements within a given interval.
- **Location**: Location is inferred from cellular ID and user answered survey. The original dataset has conveniently classified them into four category: "Home", "Work", "Elsewhere", "No Signal" on a hourly basis. We simply adopt this classification.
- **Mobility**: We infer user mobility through weighted cell-id handover counts. Handover between cellular areas carries more weight than those within an area, in order to reduce the effect of fluctuation irrelevant to mobility. Weighted sum of all handover counts within the hour is used as mobility score.
- **Social Presence**: We use the number of discovered nearby Bluetooth devices during the hour period as an indicator of social presence density.
- **Phone App Usage**: The dataset records user interaction with phone Apps, and the frequency of which is used to infer App usage context.
- **Communication Usage**: We infer communication usage from frequency of all calls and sms user made during the hour.
- **Battery Charging**: Two states "charging" and "discharging" are directly inferred. If in any instance during the hour the phone is found charging, the context is marked "charging" for this entire period.

Note that different inference methods can be used for obtaining context history of diverse accuracy and granularity. Other interested contexts can also be included. To simplify the demonstration we use some of the basic inference and settle with relatively coarse granularity, nonetheless find it adequate for evaluating the effectiveness of our scheme. In this work we focus on learning models from discrete dataset, therefore all inferred contexts are discretized with technique introduced in [18], except for **Time**, **Location**, and **Battery** which are already discrete. All processed contexts are multinomial and formated in the form of data representation specified in Table I. In addition instance that contains missing value is discarded thus all training data is complete. However learning technique for Bayesian networks can be easily extended to support continuous and incomplete dataset.

2) Synthetic Datasets: Due to the lack of groundtruth about situations, data accuracy and granularity, it is difficult using Reality Mining dataset to obtain in-depth insight into the microscopic performance of model learning that validates our design choice. Therefore we also create a synthetic dataset to assess the learning algorithms closely and with great flexibility.

We adopt the same method described in [19] to generate the synthetic dataset. The synthetic dataset is a mixture of four major situations: "rest at home", "at work", "party at home", "commute", emulating typical real-life scenarios a user may experience daily. Same contexts as those in Reality Mining data set are studied. For each situation, a component Bayesian Network is created to encode the context correlation by artificially specifying a network structure and the parameters CPT. Context observations are generated by randomly sampling the component Bayesian Network. Overall the synthetic dataset contains a total number of 4000 observations, with a sampling proportion of 6:2:1:1 for respective situations resembling typical real life scenario. We assume that all context observations are accurate in our synthetic dataset, and methods for obtaining these observations are out of the scope of this work.

Note that we intentionally engineer the situation "party at home" as a primarily evening event, with higher social presence and mobility level while lower communication usage and charging time. With this we introduce a conflicting minor situation deviating from prominent patterns in situation "rest at home", and showcase the ability of Multinet model in handling aforementioned unbalanced multi-situation cases.

In addition we design another type of synthetic data with more sense of realistic features. We mix data of two different users from Reality Mining dataset, thus creating a synthetic dataset that includes patterns of both users. Behavior patterns generally deviate more significantly across users than for a same user. They are more imbalanced in training, and some of them can be conflicting. With these two synthetic datasets we can show the benefit of Multinet models in discovering and modeling latent user patterns from mixed training data.

3) Claim Fabrication: There is no real data of fabricated context claims available publicly. To simulate the attack and evaluate the verification performance, we artificially generate fabricated datasets that consist of both genuine and fabricated claims.

First we create a dataset of seeds by randomly sub-sampling 10% out of the original data apart from training data. Therefore the seeds consist of authentic context claims. For each of the claims in the seed dataset, the adversary may decide probabilistically if a fabricate should be generated, based on a probability parameter P_{honest}. $P_{honest} = 0.4$ means there is a 40% chance that current claim will be kept intact. Lower the probability, more likely the fake claims appear. If the adversary decides to fake a claim, he will then randomly

choose one primary context as his target according to a predefined distribution over all contexts, $P_{interest}$. $P_{interest}$ specifies the likelihood of each context to be targeted by the fabrication. Upon selection, the adversary modifies the value of primary context to another valid value, thus lies about the true observation. The rest of contexts in the claim are treated as auxiliary contexts. Some of them can be intentionally removed to emulate the case of a failure in the underlying context acquisition system, or an attack in which the user suppresses the revelation of some of the contexts. We use $N_{missing}$ to control the number of auxiliary contexts will be removed, and a distribution $P_{missing}$ to determine the likelihood of which of them to be removed.

By varying these parameters, we can create fabricated dataset with different configurations. The default values for the parameters are shown in Table III. P_{honest} is set to 0.4 by default. $N_{missing}$ is set to 0.

TABLE III
DEFAULT FABRICATION PARAMETERS

Context	T	L	M	S	A	C	B
$P_{interest}$	0.05	0.3	0.2	0.2	0.15	0.05	0.05
$P_{missing}$	0.05	0.15	0.15	0.15	0.1	0.1	0.1

B. Learning Context Correlation Models

There has been a number of algorithms [19], [15], [20] proposed and can be applied to our scheme, for learning Multinet models from training dataset in an unsupervised manner. In this work we adopt the approach introduced in [20] for its simplicity in implementation.

1) Learning CL Multinets: The algorithm learns a Chow-Liu Multinet (CL Mulitnet) based on classification EM (CEM) framework. The basic idea is to iteratively learn Multinet model by alternately improving component BNs and respective data partition restricted to it. We summarize the procedure below.

- Input: Training dataset $D = \{\mathbf{x}^1, ..., \mathbf{x}^N\}$; Pre-determined number of situations K;
- Output: CL Multinet model consists of K component BNs, $MN = \{< G_1, \Theta_1 >, ..., < G_K, \Theta_K >\}$.
- Initialization: Given the training dataset and K, we need to create an initial data partition $A^0 = \{A_1^0, ..., A_k^0\}, 1 \leq k \leq K$ and corresponding Multinet model. We initialize the algorithm by first running an EM clustering algorithm to its convergence, thus creating an initial partition of size K. With the initial partition, we execute the M-step described later to learn the initial Multinet structure and parameters.
- E-Step: At mth ($m \geq 1$) iteration, for each data instances $\mathbf{x}^r, 1 \leq r \leq N$ in training set, the E-step calculates the posterior probability $t_k^m(\mathbf{x}^r)$ that $\mathbf{x}^r$ belongs to a partition A_k^{m-1}, as in Equation 1.

$$t_k^m(\mathbf{x}^r) = \frac{\alpha_k^m \prod_{i=1}^n P_k^m(x_i^r|\mathbf{Pa}(x_i^r))}{\sum_{k=1}^K \alpha_k^m \prod_{i=1}^n P_k^m(x_i^r|\mathbf{pa}(x_i^r))} \quad (1)$$

  , where x_i^r is the ith context observation in the instance and $\mathbf{Pa}(x_i^r)$ represents its parent set; α_k^m is a prior probability of variable H taking value k, associated with partition A_k^{m-1} and corresponding component BN; The product term can be calculated by Bayesian inference with the component BN parameters.
- C/S-step: In this step the algorithm creates a new partition A_k^m according to the posterior probability calculated with Equation 1. Specifically in C-step $\mathbf{x}^r$ is assigned to a partition A_k^m that provides the maximum posterior probability. To avoid converging at local maximum, C-step can be replaced by a stochastic S-step that assigns $\mathbf{x}^r$ to A_k^m with probability $t_k^m(\mathbf{x}^r)$. In our implementation we run S-step for 5 iterations then run C-step till convergence.
- M-step: The M-step learns component BNs of the Multinet by maximizing a scoring function the CML criterion:

$$CML = \sum_{k=1}^K \sum_{\mathbf{x}^r \in A_k^m} log \prod_{i=1}^n P_k^{m+1}(x_i^r|\mathbf{pa}(x_i^r)) + \sum_{k=1}^K |A_k^m| log \alpha_k^{m+1} \quad (2)$$

  All K partitions are disjoint therefore each term in the sum can be maximized separately with respect to the kth partition and component BN. The first sum is the maximum log likelihood of component BNs given data. In our implementation we choose minimum weight spanning tree (MWST) search to learn the component BN structure that maximize the term, and computes the maximum a posteriori (MAP) parameters for the learnt structures. The second sum is the prior distribution over H that can be simply maximized by setting:

$$\alpha_k^{m+1} = \frac{|A_k^m|}{n} \quad (3)$$

2) Determining Number of Situations: Before we start learning the Multinet model, the number of situations K needs to be determined first. This problem is common in many unsupervised parametric algorithms. A typical solution is to start with $K = 1$ and increase it by 1 each time, learn the Multinet model with the K value and records the resulting CML score. The incrementation stops when the improvement in CML score between two consecutive K values is less than certain threshold.

We notice that the CML score is largely affected by the final data partition, which is evolved from the initial partition. The choice of K will have bigger impact to CML score from choosing the better initial partition, than the learning process can improve from the initial partition. To relief the need of performing the full Multinet learning all the time, we propose a heuristic similar to standard approach for determining K in EM clustering. To estimate K, we perform EM clusterings on the original dataset with increasing K values, and assess the likelihood scores till its improvement is converged thus choose the best K and initial clusters. However to avoid learning from sparse data partition, if a chosen K results in any cluster having size smaller than 100 we dismiss it and choose the second best K instead and so on. The check is repeated until all clusters have size larger than 100.

C. Verifying with Context Correlation Models

With a context claim and the learnt context correlation model, we verify the integrity of primary context, X, by calculating the gain or loss in posterior probability given evidence, $\mathbf{E}$, the observed auxiliary contexts. A gain should indicate that the evidence is corroborating the claim, otherwise a loss for contradicting it.

$$P(X|\mathbf{E}) \propto P(X, \mathbf{E}) = \sum_H P(X, \mathbf{E}, H) = \sum_H P(X|\mathbf{E}, H) P(\mathbf{E}|H) P(H) \quad (4)$$

First we calculate the posterior probability following Equation 4. Calculation of $P(X|\mathbf{E}, H = k)$ is straightforward

using Bayesian inference with kth component BN given the evidence. $P(\mathbf{E}|H = k)$ can be estimated as the likelihood of evidence with kth component BN, which constitutes a process of model selection. Probability $P(X)$ with no evidence is marginalized over other contexts and component BNs. The gain between $P(X|\mathbf{E})$ and $P(X)$ is assessed. In our implementation, if the primary context reported in the claim produces highest gain in probability given evidence among all possible values, we consider the claim to be truthful. Otherwise the claim is deemed positive for fabrication.

IV. Evaluation

A. Experiment and Evaluation Settings

We implement the system components by modifying the Bayes Net Toolbox for Matlab [21], which offers a number of useful BN learning algorithms, including the MWST search algorithm and MAP learning we used, as well as inference engines. Dataset and fabricated claims are constructed as described before, and read as matlab matrix. For a comparative study we also implement a counterpart that learns generic BN model by performing a greedy structure search and the MAP parameter learning. Verification with generic BN model is the same that claim is valid if primary context inference produces highest posterior probability gain.

For each user, we randomly generate 10 seed datasets, and based on which 10 sets of fabricated claims used for a evaluation task will be generated. Different fabricated datasets for respective evaluation task are generated by varying the fabrication parameters as earlier described. Average result from them is reported as final result of the task.

For each claim in fabrication dataset, we use the learnt models to assess the probability of the claim is a fake. The verification component reports positive if it determines the claim is fabricated, negative if intact. Performance of verification is measured by comparing the reported results with groundtruth of fabrication dataset. Standard metrics of $accuracy = \frac{TP+TN}{P+N}$, $precision = \frac{TP}{TP+FP}$, and $recall = \frac{TP}{TP+FN}$ are used.

B. Evaluation Results

In this section we first show the overall performance over multiple users in order to demonstrate the effectiveness of our scheme in context verification. Subsequently we conduct case studies on selected users to investigate in detail the effect of different parameter configurations.

1) Overall Performance: In this experiment we generate fabricated datasets using aforementioned default parameters, thus creating fake claims with mixed choices of primary contexts and fully observed auxiliary contexts which represents common behavior of adversary. We apply the verification scheme on the fabricated datasets with learnt MN models, as well as the BN model for comparison. We plot the result in Figure 3. Performance varies for different users. Reasonable average accuracy over 87% is achieved, suggesting that the learnt model can be used to efficiently verify if a claim is fabricated. Specifically high recall indicates on average over 97% fabricated claims are successfully detected. The precision score suggests acceptable amount of legitimate claims are mistaken as false, which should not be much an irritation to users. Compared to BN model we observe a 17% improvement in average accuracy, and 14% improvement in precision. It confirms that dedicated model should be used to represent latent situations and conduct verification in order to maintain acceptable accuracy to verification service.

From the user survey we learn most users in the experiment live a regular schedule, with the exception of travel which however is largely discounted due to missing records. The regularity contributes a lot to the overall performance as it facilitates the accurate pattern extraction through graphical model. Based on the information from survey, we found both models performs relatively well when a user lives a highly regular, low entropy life. However for user who is not, such as user 38 who identifies his regular as "not at all", MN model shows large improvement over BN model.

2) Performance by Choice of Primary Context: One of the advantages of modeling correlation with graphic models is that the learnt model can be conveniently used to infer any arbitrary node given the evidence. This property provides us a compact framework for verification against an abstract service provider who is interested in different types of contexts. Therefore we would like to examine how such an universal model performs with respect to individual primary contexts. In each test, we alter the $P_{interest}$ distribution to select one particular context as the primary context to fabricate, and evaluate the effectiveness of learnt model in verifying by types of contexts. We iterate through all contexts for each user, and show the results for User 37 in Figure 4. Similar trend is observed in other users' results. We can find although our scheme produces varying results for verifying different primary contexts, the performance stays relatively high across all the choices. For any given primary context, the MN model consistently offers better performance compared to the generic BN model. The BN model performs surprisingly poorly when verifying **Battery** context for many users. After examining the leanrt BN we believe the reason is that high imbalance in training data for these contexts leads to failure of correctly modeling correlation between them and other contexts. Whereas MN model mitigates the problem by introducing finer models for different situations, thus maintains stable performance across contexts.

3) Effect of Conflicting Situations and Minor Patterns: In this experiment we further demonstrate with more focus the effect of conflicting situations and minor patterns to the verification performance, by a comparison between MN and generic BN. We use the synthetic dataset where a minor pattern "party at home" is overwhelmed by predominant "rest at home". Both MN and generic BN models are learnt from complete data. Then we test them with fabricated claims generated from "party" situation. Claims in the general case is generated with default $P_{interest}$ and P_{honest}, but no missing auxiliary context introduced. Moreover claims for five different primary contexts are generated and studied individually.

Results shown in Figure 5 is as expected as we find that generic BN has inferior performance for all cases. The reason is that it sometimes confuses the "party" pattern with fabrication as it deviates from the dominant pattern encoded in the single BN model. The performance is improved when MN model is used to distinguish the two situations, and build models respectively. Particularly substantial improvement in location and mobility context concurs with our design that the two situations has relatively similar observations in these two contexts while drastically diverse in others. Therefore when they are attested by auxiliary contexts conflicting in two

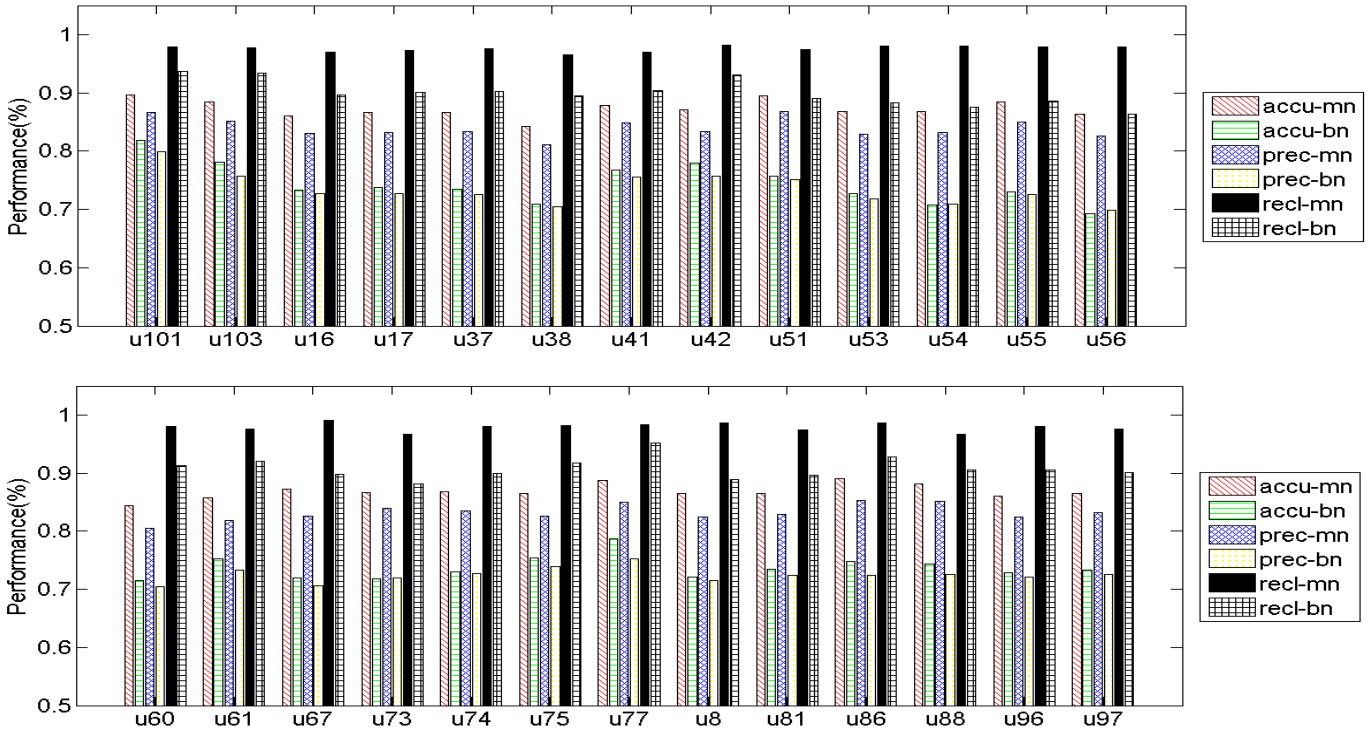
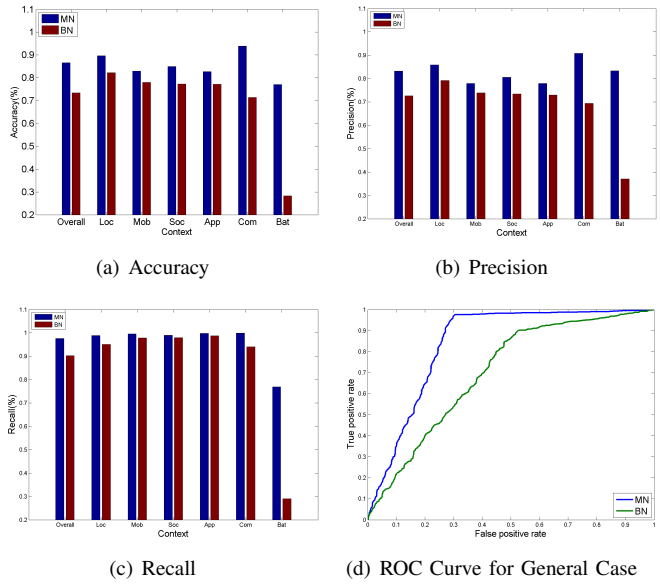


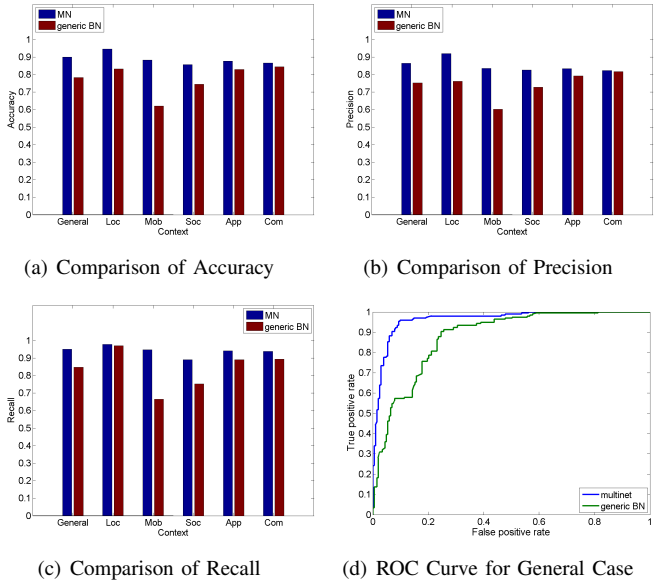
Fig. 3. Overall Performance across Users



(a) Accuracy (b) Precision

(c) Recall (d) ROC Curve for General Case

Fig. 4. Performance by Primary Context (User 37)



(a) Comparison of Accuracy (b) Comparison of Precision

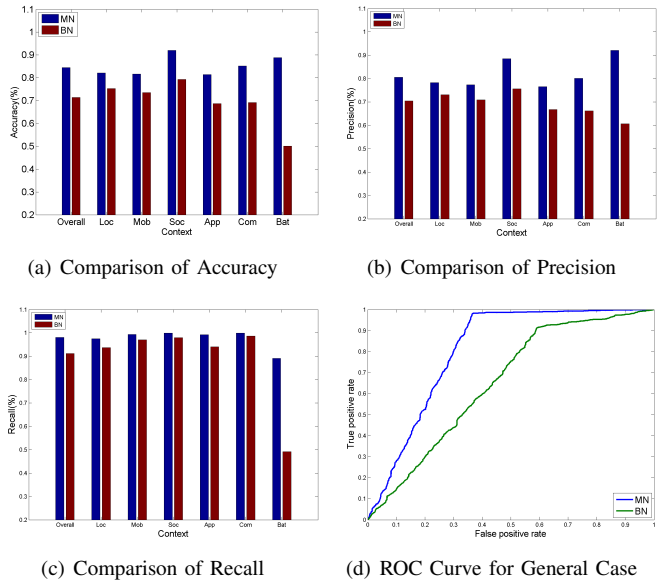
(c) Comparison of Recall (d) ROC Curve for General Case

Fig. 5. Performance with Conflicting Situations (Synthetic Data Artificial)

situations, BN model gets confused while MN model weights the correct component model and successfully verifies.

Results for the mixed synthetic dataset is shown in Figure 6. It agrees with the observation that MN model outperforms BN model, especially for contexts such as **Battery**, **Communication** which minor patterns affect most.

4) Impact of Auxiliary Contexts: The quality of auxiliary contexts will naturally affect the performance of verification. Values for certain contexts may be missing from auxiliary contexts for not been observed at the time or intentionally withheld. In this experiment, we emulate this case and try to find out how will the missing auxiliary contexts affect the performance. We fabricate with fixed primary context **Location** and P_{honest} set to 0.1 so that large quantity of fake **Location** claims are generated. Based on the generated claims, we vary $N_{missing}$ from 0 to 5, and remove respective number of auxiliary contexts that iterates all possible combinations. The result for a randomly selected user 81 of one set of missing combinations is shown in Figure 7. It is interesting to find that the two models exhibit different patterns of performance degradation. The BN model degrades only when certain contexts are removed. Since BN model suggests that a context is conditionally independent of others conditioned on its parents. Removing those independent auxiliary contexts will not affect the inference result, while large performance degradation otherwise. It is an inherent drawback of modeling



(a) Comparison of Accuracy (b) Comparison of Precision

(c) Comparison of Recall (d) ROC Curve for General Case

Fig. 6. Performance with Conflicting Situations (Synthetic Data Mixed)

with single structure as for some claims the model considers the removed contexts essential even if they are not. The MN model degrades gradually as more auxiliary contexts are removed, and it outperforms the BN model. The increasing error comes from unable to accurately assign weight to situations given the incomplete evidence, as well as the inference error with respective situation model. However we can see that the performance is surprisingly stable despite the varying missing value. The reason is that the weight calculation can take into account all remaining auxiliary contexts to pursue best possible results. In addition when inferring with individual situation model, the error is in fact amortized across different model structures. Therefore the average case performance stays stable.

V. Related Work

A. Sensor and Context Authenticity Attestation

To address the challenges in trustworthy sensor data collection and processing, some existing solutions propose deployment of trustworthy infrastructure [22] or relying on a collective of other participants [8] when dealing with certain types of context namely the location. A more general solution usually resorts to signing the sensor data right after it is sensed at the sensing device, either by deploying a dedicated sensing and signing hardware [6], [23] or exploiting Trusted Platform Module (TPM) that already implemented in some commodity mobile devices [24]. These schemes encapsulate the acquirement process of sensor data in a trusted and verifiable environment created by the trusted computing base, and thus protect the data from malicious tampering even if the operating system and user application are compromised. Other schemes [25], [26], [27] are proposed to attest programs or codes used to process raw sensor data, by leveraging TPM as well. More recently, systems [28], [7] are implemented to attest both sensor data and local processing of the data. Despite the sound security property the systems provide, the requirement of operating in trusted computing environment still poses substantial latency, computational, and deployment overhead. These approaches target at protecting sensor and context authenticity at the data source, suggesting that any data must be attested when created for the protection to be in effective. For instance as evaluated in [7] attesting a sensor reading may introduce latency up to 40 milliseconds, exerting high pressure on application that continuously monitors sensors.

Instead, we choose a different approach that uses correlation between contexts for the attestation. The core functionality does not require TPM. Most of the computation can be offloaded to a powerful cloud server, while the inference is designed to be efficient. Depending on the application, our scheme can also work alongside with the existing solutions to achieve maximum security guarantee.

B. Context Correlation Modeling

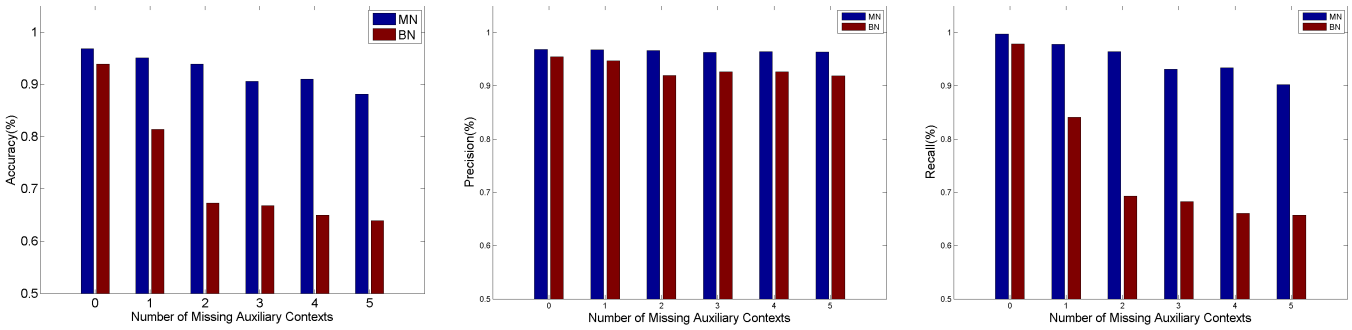
Various data mining techniques have been employed to model relations among sensor and context data. One line of research[29], [30] tends to use the models to resolve conflicting information collected from different data sources such as sensor nodes or other participants. Instead of verifying different versions of a piece of information, we use information collected from secondary domains from the same source, to verify the primary domain information. In doing so we minimize the need of other sources for providing alternative versions, which may not be always available for many types of applications.

In this regard we share the same idea with [31], [10], [11], [32]. In [31] and [10], decision tree and association rules are used respectively to model correlation among sensors in a task to reduce energy consumption. Inference with these models are rather deterministic and cannot deal with uncertainty and conflicts in sensor data well if applied to context verification. Probabilistic graphical models provides a concise framework that integrates context uncertainty nicely, together with a straightforward inference algorithm based on Bayesian rules. In [11] the authors exploit the Dynamic Bayesian Networks for modeling correlation. However as analyzed in Section II it cannot address the particular issue with conflicting situations. In our work a Bayesian Multinets model is adopted to cope with this problem. In [32] a mixture model is proposed to discover behavior pattern in different situations, but only time and location are studied.

VI. Conclusion and Future Work

In this paper we propose a novel approach for verifying the authenticity of user-reported context claims, by assessing the correlation among co-observed contexts. We present a preliminary study focusing on learning a Multinet model from user context history that encodes context correlations existing in various situations. Using the learnt model to verify fabricated context claims produces encouraging results with Reality Mining dataset, demonstrating the feasibility of the scheme.

In the future, we would like to further validate the model with real life dataset of larger context sets and finer context granularity, which can provide us in-depth insight into the mechanism of inference with context correlation. Based on the new understanding, advanced processing and modeling techniques can applied to improve the accuracy of learnt model in representing user behavior pattern. For instance a feature selection phase can be included when pool of contexts are larger to minimize the interference irrelevant auxiliary



(a) Accuracy (b) Precision (c) Recall

Fig. 7. Impact of Missing Values in Auxiliary Contexts (User 81)

contexts. Dynamic Multinets [33] can also be used to enhance the modeling for temporal correlations.

Another aspect we intend to look into is designing and evaluating attestation schemes resilient to powerful adversaries who also have certain level of knowledge about context correlation, and study the scenario where the training dataset is contaminated with fabricated context history. To deter a knowledgeable adversary who can exploit context correlation and tamper with some of the auxiliary contexts, one possible solution is to include more context related to phone usage which are more difficult to fake without affecting user experience. Dynamic Multinets that evaluates temporal correlations among consecutive context claims may assist in defending powerful adversary. In addition a trust framework can be deployed to assess the risk and provide feedback to verification decision in the presence of powerful adversary.

REFERENCES

[1] Foursquare, *Foursquare*, 2014, https://foursquare.com/.

[2] Progressive, *Snapshot*, 2014, http://www.progressive.com/auto/snapshot/?vanity=true/.

[3] Nike, *Nike+*, 2014, https://secure-nikeplus.nike.com/plus/products/gps_app/.

[4] W. He, X. Liu, and M. Ren, "Location cheating: A security challenge to location-based social network services," in *ICDCS'11*, June 2011, pp. 740–749.

[5] openintents, *SensorSimulator*, 2014, https://code.google.com/p/openintents/wiki/SensorSimulator.

[6] A. Dua, N. Bulusu, W.-C. Feng, and W. Hu, "Towards trustworthy participatory sensing," in *USENIX*, ser. HotSec'09, Berkeley, CA, USA, 2009, pp. 8–8.

[7] H. Liu, S. Saroiu, A. Wolman, and H. Raj, "Software abstractions for trusted sensors," in *ACM*, ser. MobiSys '12, New York, NY, USA, 2012, pp. 365–378.

[8] X. Wang, J. Zhu, A. Pande, A. Raghuramu, P. Mohapatra, T. Abdelzaher, and R. Ganti, "Stamp: Ad hoc spatial-temporal provenance assurance for mobile users," in *ICNP'13*, Oct 2013, pp. 1–10.

[9] A. Padovitz, S. W. Loke, and A. Zaslavsky, "Towards a theory of context spaces," in *Pervasive Computing and Communications Workshops, 2004.*, March 2004, pp. 38–42.

[10] S. Nath, "Ace: exploiting correlation for energy-efficient and continuous context sensing," ser. MobiSys '12. ACM, 2012, Conference Paper, pp. 29–42.

[11] A. Parate, M.-C. Chiu, D. Ganesan, and B. M. Marlin, "Leveraging graphical models to improve accuracy and reduce privacy risks of mobile sensing," ser. MobiSys '13. ACM, 2013, Conference Paper, pp. 83–96.

[12] H. Höpfner and K. Sattler, "Cache-supported processing of queries in mobile DBS," in *Database Mechanisms for Mobile Applications, Workshop by the GI-Arbeitskreis*, 2003, pp. 106–121.

[13] J. Pearl, *Causality: Models, Reasoning, and Inference*. New York, NY, USA: Cambridge University Press, 2000.

[14] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Mach. Learn.*, vol. 29, no. 2-3, pp. 131–163, Nov. 1997.

[15] N. Friedman, "The bayesian structural em algorithm," in *Proceedings of UAI'98*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998, pp. 129–138.

[16] D. Geiger and D. Heckerman, "Knowledge representation and inference in similarity networks and bayesian multinets," *Artif. Intell.*, vol. 82, no. 1-2, pp. 45–74, Apr. 1996.

[17] N. Eagle and A. (Sandy) Pentland, "Reality mining: Sensing complex social systems," *Personal Ubiquitous Comput.*, vol. 10, no. 4, pp. 255–268, Mar. 2006.

[18] O. Alata, C. Olivier, and Y. Pousset, "Law recognitions by information criteria for the statistical modeling of small scale fading of the radio mobile channel," *Signal Process.*, vol. 93, no. 5, pp. 1064–1078, May 2013.

[19] B. Thiesson, C. Meek, D. M. Chickering, and D. Heckerman, "Learning mixtures of dag models," in *Proceedings of UAI'98*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998, pp. 504–513.

[20] D. T. Pham and G. A. Ruz, "Unsupervised training of bayesian networks for data clustering," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, vol. 465, no. 2109, pp. 2927–2948, 2009.

[21] K. Murphy, *Bayes Net Toolbox for Matlab*, 2014, https://code.google.com/p/bnt/.

[22] S. Saroiu and A. Wolman, "Enabling new mobile applications with location proofs," in *Proceedings of the 10th Workshop on Mobile Computing Systems and Applications*, ser. HotMobile '09. New York, NY, USA: ACM, 2009, pp. 3:1–3:6.

[23] ——, "I am a sensor, and i approve this message," in *Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications*, ser. HotMobile '10. New York, NY, USA: ACM, 2010, pp. 37–42.

[24] P. Gilbert, L. P. Cox, J. Jung, and D. Wetherall, "Toward trustworthy mobile sensing," in *Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications*, ser. HotMobile '10. New York, NY, USA: ACM, 2010, pp. 31–36.

[25] J. M. McCune, B. J. Parno, A. Perrig, M. K. Reiter, and H. Isozaki, "Flicker: An execution infrastructure for tcb minimization," in *Proceedings of the 3rd ACM SIGOPS/EuroSys European Conference on Computer Systems 2008*, ser. Eurosys '08. New York, NY, USA: ACM, 2008, pp. 315–328.

[26] M. Nauman, S. Khan, X. Zhang, and J.-P. Seifert, "Beyond kernel-level integrity measurement: Enabling remote attestation for the android platform," in *Proceedings of the 3rd International Conference on Trust and Trustworthy Computing*, ser. TRUST'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 1–15.

[27] F. B. Schneider, K. Walsh, and E. G. Sirer, "Nexus authorization logic (nal): Design rationale and applications," *ACM Trans. Inf. Syst. Secur.*, vol. 14, no. 1, pp. 8:1–8:28, Jun. 2011.

[28] P. Gilbert, J. Jung, K. Lee, H. Qin, D. Sharkey, A. Sheth, and L. P. Cox, "Youprove: Authenticity and fidelity in mobile sensing," in *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '11. New York, NY, USA: ACM, 2011, pp. 176–189.

[29] X. L. Dong, L. Berti-Equille, and D. Srivastava, "Truth discovery and copying detection in a dynamic world," *Proc. VLDB Endow.*, vol. 2, no. 1, pp. 562–573, Aug. 2009.

[30] D. Wang, L. Kaplan, H. Le, and T. Abdelzaher, "On truth discovery in social sensing: A maximum likelihood estimation approach," in *ACM*, ser. IPSN '12, New York, NY, USA, 2012, pp. 233–244.

[31] A. Deshpande, C. Guestrin, W. Hong, and S. Madden, "Exploiting correlated attributes in acquisitional query processing," in *IEEE*, ser. ICDE '05, Washington, DC, USA, 2005, pp. 143–154.

[32] J. Zheng and L. M. Ni, "An unsupervised framework for sensing individual and cluster behavior patterns from human mobile data," in *Proceedings of UbiComp '12*. New York, NY, USA: ACM, 2012, pp. 153–162.

[33] J. Bilmes, "Dynamic bayesian multinets," in *Proceedings of UAI '00*, 2000, pp. 38–45.