Privacy-Preserving Mobility Monitoring Using Sketches of Stationary Sensor Readings

Conference Paper · January 2013

DOI: 10.1007/978-3-642-40994-3_24

Michael Kamp
Monash University (Australia)

Mario Boley
Fritz Haber Institute of the Max Planck Society

Privacy-Preserving Mobility Monitoring using Sketches of Stationary Sensor Readings

Michael Kamp, Christine Kopp, Michael Mock, Mario Boley, and Michael May
{michael.kamp, christine.kopp, michael.mock, mario.boley,
michael.may}@iais.fraunhofer.de

Fraunhofer IAIS, Schloss Birlinghoven
St. Augustin, Germany

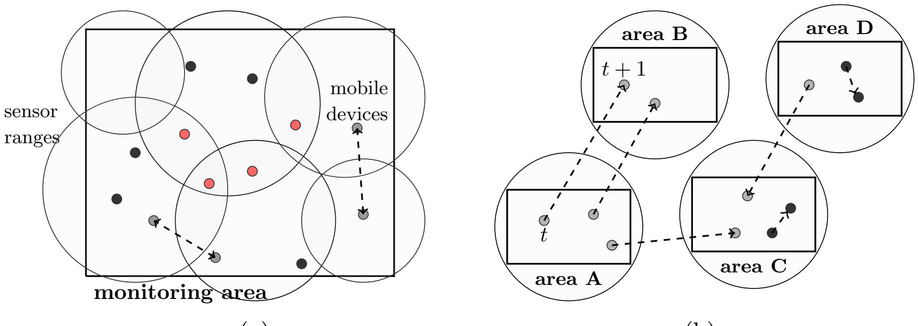
Abstract. Two fundamental tasks of mobility modeling are (1) to track the number of distinct persons that are present at a location of interest and (2) to reconstruct flows of persons between two or more different locations. Stationary sensors, such as Bluetooth scanners, have been applied to both tasks with remarkable success. However, this approach has privacy problems. For instance, Bluetooth scanners store the MAC address of a device that can in principle be linked to a single person. Unique hashing of the address only partially solves the problem because such a pseudonym is still vulnerable to various linking attacks. In this paper we propose a solution to both tasks using an extension of linear counting sketches. The idea is to map several individuals to the same position in a sketch, while at the same time the inaccuracies introduced by this overloading are compensated by using several independent sketches. This idea provides, for the first time, a general set of primitives for privacy preserving mobility modeling from Bluetooth and similar address-based devices.

1 Introduction

Advanced sensor technology and spread of mobile devices allows for increasingly accurate mobility modeling and monitoring. Two specific tasks are crowd monitoring, i.e., counting the number of mobile entities in an area, and flow monitoring between locations, i.e., counting the number of entities moving from one place to another within a given time interval. [1] Both have several applications in event surveillance and marketing [10, 16]. Moreover, matrices containing the flow between every pair of locations (origin-destination, or OD-matrices) are an important tool in many GIS applications, notably traffic planning and management [3].

Today's sensor technologies such as GPS, RFID, GSM, and Bluetooth have revolutionized data collection in this area, although significant problems remain to be solved. One of those problems are privacy concerns. They mandate that, while the count of groups of people can be inferred, inference on an individual

[1] In this paper, we use the term 'flow' always as a short-hand for 'flow between two or more locations'.



(a) (b)

Fig. 1: The two mobility modeling tasks addressed in this paper: (a) *crowd monitoring* and (b) *flow monitoring*.

person remains infeasible. Directly tracing IDs through the sensors violates this privacy constraint, because the amount of information stored allows for linking attacks [12]. In such an attack, sensor information is linked to additional knowledge in order to identify a person and infer upon her movement behavior. Hence, application designers have to design and use new, privacy preserving methods.

The contribution of this paper is to provide a general set of primitives for privacy-preserving mobility monitoring and modeling using stationary sensor devices. Following the *privacy-by-design* paradigm [17, 21], we present a method that stores just enough information to perform the desired inference task and discards the rest. Thereby, privacy constraints are much easier to enforce.

Technically, the method we propose is based on Linear Counting sketches [26], a data structure that allows to probabilistically count the distinct amount of unique items in the presence of duplicates. Linear Counting not only obfuscates the individual entities by hashing, but furthermore provides a probabilistic form of k-anonymity. This form of anonymity guarantees that, by having access to all stored information, an attacker cannot be certain on a single individual but can at most infer upon k individuals as a group. Furthermore, Linear Counting is an efficient and easy to implement method that outperforms other approaches in terms of accuracy and privacy on the cost of higher memory usage [19].

The rest of the paper is structured as follows. After discussing related work in section 2, we describe the application scenarios in section 3. In section 4 we present our extension to the linear counting method and give a theoretical analysis of the error. Subsequently, we analyze the privacy of our method in section 5. In section 6 we conduct extensive experiments on the accuracy of Linear Counting and flow estimation under different privacy requirements to test our approach. These experiments have been carried out on a a real-world simulation. Section 7 concludes with a discussion of the results and pointers to future directions.

2 Related Work

The basic tool we are using in this paper are sketches (see Cormode et al. [9] for a good general introduction). Sketches are summaries of possibly huge data collections that, on the one hand, discard some information for the sake of space efficiency or privacy, but that, on the other hand, still contain enough information to restore the current value of certain variables of interest. Sketches are a very universal tool and have been successfully applied for inferring heavy hitters, moments of a distribution, distinct elements, and more. The general idea of using sketches for privacy is described in Aggarwal and Yu [1] and Mir et al. [20]. The first relates the privacy of a sketch to the variance of the estimator. The latter discusses privacy paradigms that go beyond differential privacy [11] and address security as well. They use various techniques for achieving privacy, notably adding noise. In our approach, we do not employ noise as a source of privacy. However, due to the probabilistic nature of our method, noise has no excessive impact on the accuracy. Hence, adding noise to improve privacy can be combined with our method.

The crucial task in this paper is to count the number of distinct objects at a location (see Gibbons [14] for a general overview on approaches for this problem). The method discussed in our paper, Linear Counting, is first described in [26]. This method can be seen as a special case of Bloom filters for counting [6]. So far, it has received relatively little attention, because it is not as space efficient as log-space approaches such as FM sketches, and traditionally sketches have mainly been used to provide short summaries of huge data collections. However, in the context of privacy preservation in the mobility modeling scenarios of this paper, space is not so much an issue as is *accuracy.* To this end, recently very positive results are reported for comparisons of Linear Counting with FM sketches and other methods [19, 15]. Especially for smaller set sizes that appear in mobility mining, Linear Counting has often an advantage in terms of accuracy. Hence, for our scenario, it is a promising choice.

The idea of using Linear Counting for stationary sensors, specifically Bluetooth measurements, recently has also been described by Gonçalves et al. [15] and similar work for mobile devices is reported in Bergamini et al. [4]. However, here, for the first time, we describe extensions of the basic Linear Count sketches that can be used to monitor flows between locations as well as to compensate for the precision loss incurred for raising privacy.

Our approach for tracking flows is based on the ability to compute the intersection of sketches. A general method for computing general set expressions from sketches is described in Ganguly et al. [13]. Though highly general, the approach is ruled out for our application because the scheme requires to store information at the coordinator which could be used for identifying persons and thus is not privacy aware—it should be noted that it was not build for that purpose.

3 Application Scenarios

In the following, the two application scenarios in mobility monitoring covered by this paper are described. The general setup in both applications is using stationary sensor devices centralizing their sensor readings at a coordinator.

In general, a well-studied approach to monitoring people in an area is to use sensors that count the number of mobile devices in their sensor radius [14, 15], such as Bluetooth, or GSM scanners. From the amount of mobile devices, the actual amount of people can be accurately estimated by assuming a stable average fraction of people carrying such a device [18].

A stationary sensor S scans periodically for devices in its sensor range. Each device is identifiable by its address a from a **global address space** $\mathcal{A}$. The stream of **sensor readings** of a sensor S is defined by $\mathcal{R}_S \subseteq \mathcal{A} \times \mathbb{R}_+$ where $(a,t) \in \mathcal{R}_S$ means that S has read a device address a at time stamp t. For a measurement interval $T = [b,e] \subseteq \mathbb{R}_+$, the readings of sensor S in this time interval is denoted by $\mathcal{R}_S^T = \{a \in \mathcal{A} | \exists t \in T : (a,t) \in \mathcal{R}_S\}$. In both application scenarios, the sensor readings are evaluated to solve the application-specific problem.

3.1 Crowd Monitoring

In major public events, such as concerts, sport events or demonstrations, continuously monitoring the amount of people in certain areas is a key tool for maintaining security. It is vital for the prevention of overcrowding as well as for allocating security and service personnel to the places they are needed most.

To monitor an area using stationary sensor devices, a set of sensors $\mathcal{S} = \{S_1, ..., S_k\}$ is distributed over the area such that the union of their sensor ranges covers the complete area (see Fig. 1(a)). For a single sensor S, the **count distinct** of unique entities that have been present in its range during a time interval T is $\left|\mathcal{R}_S^T\right|$. The task is then to continuously monitor at a central site the count distinct of the **union** of all sensor ranges. That is, we aim to monitor $|\mathcal{R}_{S_1}^{T_i} \cup ... \cup \mathcal{R}_{S_k}^{T_i}|$ for consecutive measurement intervals $T_0, T_1, T_2 ...$ of a fixed time resolution.

Note that the problem of duplicates in the sensor readings cannot be avoided in practice because of several reasons: Covering an area with circular sensor ranges requires overlap and the radius of each sensor range cannot be accurately estimated beforehand. Furthermore, entities can move between sensor ranges during a measurement interval. Thus, independent of the specific application design, summing the distinct counts of individual sensor readings usually overestimate the true number of devices. Without privacy restrictions, this problem can be solved by centralizing the read device addresses or a unique hash of them to eliminate duplicates. However, when addresses or unique identifiers are centralized, devices can be tracked and linked to real persons, thereby violating common privacy constraints.

To solve this problem, we use **Linear Counting** that has been introduced as an accurate and privacy preserving method for estimating the number of distinct items in the presence of duplicates. Linear counting is a sketching technique, i.e., the vector of sensor readings is compressed to a lower dimensional vector, the so called sketch, in such a way that a desired quantity can still be extracted sufficiently accurate from the sketch. The linear count sketch maintains privacy by deliberately compressing the sensor readings with a certain amount of collisions, such that a deterministic inference from a sketch to an address is impossible. A detailed analysis of the privacy aspect is provided in section 5. The distinct amount of people in an area covered by several overlapping sensors is estimated by combining the individual sketches to a sketch of the union of sensor ranges. The method is described in section 4.1. For that, only the sketches have to be stored and send to the coordinator so that privacy is preserved at a very basic level. This provides two general primitives for monitoring the amount of entities in an area: linear count sketches for privacy-preserving estimation of the **count distinct** of mobile entities in a sensor radius and estimation of the distinct count of entities in an area defined as the **union** of sensor ranges.

3.2 Flow Monitoring

The flow of mobile entities, such as pedestrians, cars or trains, is a key quantity in traffic planning. Streets, rails and public transportation networks are optimized for handling these flows. For a given set of locations, all flows between the locations can be combined in the form of an origin-destination matrix. These matrices are an important tool in traffic management [2].

Flows of mobile entities between a set of locations can be estimated using stationary sensors [3]. For that, given a set of locations of interest, a sensor is placed at each location (e.g., see Fig. 1(b)). For a given time period, the flow between two locations denotes the amount of mobile entities that have been present at one location at the beginning of that time period and present at the other location at the end of the period. An entity is present at a location, if it is staying within the range of the sensor placed at that location. Thus, given a time interval at the beginning of the period, T_b, and one at the end, T_e, the **flow** between two sensors S and S' is defined as $v\left(S, S'\right) = |\{a \in \mathcal{A} \mid a \in \mathcal{R}_S^{T_b} \wedge a \in \mathcal{R}_{S'}^{T_e}\}|$. For convenience, we assume that sensor ranges do not overlap. In the case of overlap, this notion of flow has to be extended so that the number of mobile devices that stayed in the intersection is handled separately.

The existing approaches to flow monitoring with stationary sensors rely on tracing unique identifiers through the sensor network. Hence, identifying and tracking a specific device, i.e., a specific person, is possible in monitoring systems as soon as the identifier can be linked to a person. Again, this violates common privacy restrictions. In order to monitor flows in a privacy-preserving manner using linear count sketches, the definition of flow is modified to be able to express the flow as the intersection of sensor readings of different time intervals. Therefore, let T_b and T_e be disjoint time intervals as in the aforementioned definition of flow, then the flow can be expressed as $v\left(S, S'\right) = |\mathcal{R}_S^{T_b} \cap \mathcal{R}_{S'}^{T_e}|$. In section 4.2, we show how local linear count sketches can be combined in a privacy-preserving manner to estimate the size of an intersection.

The extension to paths of arbitrary length is straightforward. Given three consecutive, disjoint time intervals T_1, T_2, T_3, the flow on a path $S_1 \rightarrow S_2 \rightarrow S_3$ can be represented as $|\mathcal{R}_{S_1}^{T_1} \cap \mathcal{R}_{S_2}^{T_2} \cap \mathcal{R}_{S_3}^{T_3}|$. However, the accuracy of the flow estimation is highly dependent on the size of the intersection compared to the original sets. Moreover, the accuracy drops drastically in the number of intersections, thereby limiting the length of paths that can be monitored. To boost the accuracy and thereby ensure applicability, we present an improved estimator that uses a set of intermediate estimators and their mean value in section 4.3.

An OD-matrix L for a set of locations $\{l_1, ..., l_k\}$ is defined as the flow between each pair of locations, i.e., $L \in \mathbb{R}^{k \times k}$ with $L_{ij} = v\left(l_i, l_j\right)$. By placing a sensor at each location, that is, sensor S_i is placed at location l_i, an OD-matrix L can be estimated by $L_{ij} = v\left(S_i, S_j\right)$. This provides another mobility mining primitive: the estimation of flows, paths and OD-matrices based on the **intersection** of sensor readings.

4 Extending Linear Counting

In this section, we present the technical solution to the application scenarios introduced in section 3. We start by recapitulating Linear Counting sketches, which serve as fundamental tool. In particular, for the flow-monitoring it is necessary to extend this sketching technique to monitoring the size of an intersection of two or more sensor readings. For both application scenarios, privacy preservation demands that we use basic sketches at the sensors with relatively high variance in their estimates. This variance even increases when estimating intersections. In order to increase the accuracy again on the output layer, we present an improved estimator that reduces the variance by combining several independent sketches.

4.1 From Sensors to Sketches

Given the sensor readings $\mathcal{R}_S^T = \{a \in \mathcal{A} \mid \exists t \in T : (a, t) \in \mathcal{R}_S\}$ of a sensor S during a time interval T, the goal is to represent the number of distinct devices observed without explicitly storing their addresses. A problem of this kind is referred to as **count distinct problem**, which can be tackled by Linear Counting sketches [26]. They have originally been introduced to estimate the number of unique elements within a table of a relational database.

In our scenario, this means that, instead of storing all readings within a measurement interval T, a sensor just maintains a binary **sketch** $\mathrm{sk}\big(\mathcal{R}_S^T\big) \in \{0, 1\}^m$ of some fixed length m. The sketch is determined by a random **hash map** $h : \mathcal{A} \to \{0, \ldots, m-1\}$ such that the following **uniformity** property holds: for all $a \in \mathcal{A}$ and all $k \in \{0, \ldots, m-1\}$ it holds that $\mathbb{P}(h(a) = k) = 1/m$. For practical purposes this can be approximately achieved by choosing, e.g., $h(a) = ((va + w) \mod p) \mod m$, with uniform random numbers $v, w \in \mathbb{N}$ and a fixed large prime number p. Other choices of hash functions are possible (see e.g., Preneel [22]).

A sensor maintains its sketch as follows. At the beginning of the measurement interval it starts with an empty sketch $(0, \ldots, 0)$, and on every address $a \in \mathcal{A}$ read, until the end of the interval, the sketch is updated by setting the $h(a)$-th position to 1. For the whole measurement interval this results in a sketch

$$\mathrm{sk}\big(\mathcal{R}_S^T\big)\,[k] = \begin{cases} 1 & \text{, if } \exists a \in \mathcal{R}_S^T,\, h(a) = i \\ 0 & \text{, otherwise} \end{cases} \quad .$$

In our application scenarios, a **global population** of mobile entities $\mathcal{P} \subseteq \mathcal{A}$ of size $|\mathcal{P}| = n$ is monitored with a set of sensors. The size of the global population

n is an upper bound to the number of mobile entities in a sensor reading. In each sensor reading, a subset of the global population is captured, i.e., $\mathcal{R}_S^T \subseteq \mathcal{P}$, thus $|\mathcal{R}_S^T| \leq |\mathcal{P}|$, or $n_S \leq n$ (from now on we denote $\left|\mathcal{R}_S^T\right|$ as n_S).

The number of distinct addresses within a sensor reading can then be estimated based on the sketch as follows. Assume a sensor reading $\mathcal{R}_S^T$ with $\left|\mathcal{R}_S^T\right| = n_S$ and the respective sketch $\mathrm{sk}\left(\mathcal{R}_S^T\right)$. Let $\mathbf{u}_S$ denote the number of zeros in the sketch and $\mathbf{v}_S = \mathbf{u}_S/m$ the relative zero count. Now, the maximum likelihood **count estimator** for the number of distinct items n_S is $\widehat{n}_S = -m \ln \mathbf{v}_S$. Whang et al. [26] shows that the expected value, and the variance of this estimator are asymptotically well-behaved. Here, asymptotically refers to the **limit** for increasing n while the **loadfactor** $t = n/m$ and the **relative size** $c_S = n_S/n$ of S are kept constant. With this notion of limit, that we simply denote by lim for the remainder of this paper, the expected value and the variance can be expressed as

$$\lim \mathbb{E}[\widehat{n}_S] = n_S + (e^{n_S/m} - n_S/m - 1)/2 = n_S \tag{1}$$

$$\lim \mathbb{V}(\widehat{n}_S) = m\left(e^{n_S/m} - n_S/m - 1\right) \quad , \tag{2}$$

respectively. Hence, asymptotically, the estimator is unbiased and has a bounded variance. Standard concentration inequalities can be used to convert this result into probabilistic error guarantees.

In the crowd monitoring scenario, the size of the global population n can be estimated as the size of the union of all individual sensor readings. By construction of the sketches, it is possible to build a sketch of the union of sensor readings $\mathcal{R}_S^T$ and $\mathcal{R}_{S'}^T$ by combining the individual sketches with the point-wise binary OR operation (i.e., $\mathrm{sk}\left(\mathcal{R}_S^T\right)[k] \vee \mathrm{sk}\left(\mathcal{R}_{S'}^T\right)[k]$ is equal to 1 if and only if $\mathrm{sk}\left(\mathcal{R}_S^T\right)[k] = 1$ or $\mathrm{sk}\left(\mathcal{R}_{S'}^T\right)[k] = 1$). The following statement holds:

Proposition 1 (Whang et al. [26]). *Let $\mathcal{R}_{S_1}, ..., \mathcal{R}_{S_k}$ be readings of a set of sensors $S = S_1, ..., S_k$. The sketch constructed from the union of these readings can be obtained by calculating the binary or of the individual sketches. That is,*

$$sk\left(\bigcup_{i=1}^{k} \mathcal{R}_{S_i}\right) = \bigvee_{i=1}^{k} sk(\mathcal{R}_{S_i}) \quad .$$

This is already sufficient to continuously track the total number of distinct addresses in the crowd monitoring scenario: for a pre-determined time resolution, the sensor nodes construct sketches of their readings, send them to a monitoring coordinator, and start over with new sketches. As required by the application, the coordinator can then compute the estimate of distinct counts of mobile entities based on the OR-combination of all local sketches. However, for the flow-monitoring scenario we have to be able to compute the number of distinct addresses in the intersection of sensor readings. Therefore, we have to extend the Linear Counting approach.

4.2 Intersection Estimation

In the following, a method is presented for estimating the intersection of two sets using linear count sketches. The sketch of the intersection cannot be constructed from the individual sketches (note that using the binary 'and' operation on the two sketches does in general not result in the correct sketch of the intersection). Therefore, this method is based on the inclusion-exclusion formula for sets. That is, we can express the size of the intersection of two sets A, B as $|A \cap B| = |A| + |B| - |A \cup B|$. The estimator for the intersection of two sensor ranges is defined in a similar way, using the estimators for each sensor and their union. Let $\widehat{n}_S, \widehat{n}_{S'}$ denote the estimator for $\left|\mathcal{R}_S^T\right|$, respectively $\left|\mathcal{R}_{S'}^T\right|$. Let furthermore $\widehat{n}_{S\cup S'}$ denote the estimator based on the sketch of the union of the sensor readings $\mathcal{R}_S^T$ and $\mathcal{R}_{S'}^T$ as defined in proposition 1. Then the **intersection estimator** is defined as

$$\widetilde{n}_{S,S'} = \widehat{n}_S + \widehat{n}_{S'} - \widehat{n}_{S\cup S'} \tag{3}$$

It turns out that also this estimator asymptotically is unbiased and has a bounded variance. The first follows directly from the linearity of the expected value. Thus, we can note:

Proposition 2. *For a constant loadfactor t and constant fractions c_S, c'_S, the estimator $\widetilde{n}_{S,S'}$ is asymptotically unbiased, i.e.,* $\lim \mathbb{E}[\widetilde{n}_{S,S'}]/|S \cap S'| = 1$.

Furthermore, we can bound the variance of our estimator by the variance of the union estimator. This implies that resulting probabilistic error guarantees become tighter the closer the ratio $|S \cap S'|/|S \cup S'|$ is to one.

Proposition 3. *Asymptotically, the variance of the intersection estimator $\widetilde{n}_{S,S'}$ is bounded by the variance of the count estimator for the union, i.e.,* $\lim \mathbb{V}(\widetilde{n}_{S,S'}) \leq \lim \mathbb{V}(\widehat{n}_{S\cup S'})$.

Proof (sketch). For some subset $A \subseteq \mathcal{P}$ and a fixed sketch position $k \in \{0, \ldots, m-1\}$ let us denote by $p_A = \mathbb{P}(sk(A)[k] = 0)$ the probability that the sketch of A has entry 0 at position k. Due to the uniformity of the hash function h it holds that $p_A = (1 - 1/m)^{n_A}$. The limit of this probability $p_A^* = \lim p_A$ is equal to $\lim (1 - t/n)^{nc_A} = e^{-t \cdot c_A}$. The variance of the intersection estimator can be re-expressed in terms of the covariances σ of the individual count estimators:

$$\begin{aligned} \mathbb{V}(\widetilde{n}_{S,S'}) &= \mathbb{V}(\widehat{n}_S + \widehat{n}_{S'} - \widehat{n}_{S\cup S'}) \\ &= \mathbb{V}(\widehat{n}_S) + \mathbb{V}(\widehat{n}_{S'}) + \mathbb{V}(\widehat{n}_{S\cup S'}) + 2\sigma(\widehat{n}_S, \widehat{n}_{S'}) \\ &\quad - 2\sigma(\widehat{n}_S, \widehat{n}_{S\cup S'}) - 2\sigma(\widehat{n}_{S'}, \widehat{n}_{S\cup S'}) \ . \end{aligned} \tag{4}$$

In order to determine the limit of the covariances for the count estimators $\hat{n}_A$ and $\hat{n}_B$ for some arbitrary subsets $A, B \subseteq \mathcal{P}$ we can use Whang et al. [26, Eq. (8)] and the bi-linearity of the covariance

$$\begin{aligned} \lim \sigma(\hat{n}_A, \hat{n}_B) &= \sigma(m\,(tc_A - \mathbf{v}_A/p_A^* - 1), m\,(tc_B - \mathbf{v}_B/p_B^* - 1)) \\ &= m^2\sigma(\mathbf{v}_A, \mathbf{v}_B)/(p_A^* p_B^*) = m^2\sigma(\mathbf{u}_A/m, \mathbf{u}_B/m)/(p_A^* p_B^*) \\ &= \sigma(\mathbf{u}_A, \mathbf{u}_B)/(p_A^* p_B^*) \ . \end{aligned} \tag{5}$$

Let $\overline{\mathrm{sk}}(A)\,[k]$ denote the binary negation of sketch position k. The co-variances of the absolute number of zero entries $\mathbf{u}_A$ and $\mathbf{u}_B$ is

$$\begin{aligned}\sigma(\mathbf{u}_A, \mathbf{u}_B) =& \sum_{k=1}^{m} \sum_{l=1}^{m} \sigma\big(\overline{\mathrm{sk}}(A)\,[k], \overline{\mathrm{sk}}(B)\,[l]\big) \\ =& \sum_{k=1}^{m} \sigma\big(\overline{\mathrm{sk}}(A)\,[k], \overline{\mathrm{sk}}(B)\,[k]\big) + \sum_{\substack{k,l=1\\k\neq l}}^{m} \sigma\big(\overline{\mathrm{sk}}(A)\,[k], \overline{\mathrm{sk}}(B)\,[l]\big) \ \ .\end{aligned} \tag{6}$$

Let $U = A \cup B$ and $I = A \cap B$. We state without proof that the co-variances of the individual sketch positions are given by

$$\begin{aligned}\sigma\big(\overline{\mathrm{sk}}(A)\,[k], \overline{\mathrm{sk}}(B)\,[k]\big) &= p_U - p_A p_B \\ \sigma\big(\overline{\mathrm{sk}}(A)\,[k], \overline{\mathrm{sk}}(B)\,[l]\big) &= p_{A\setminus B}\, p_{A\setminus B}(1-2/m)^{|I|} - p_A p_B\end{aligned}$$

for the cases $k = l$ and $k \neq l$, respectively. From this and Eq. (6) it follows that

$$\begin{aligned}\lim \sigma(\mathbf{u}_A, \mathbf{u}_B) &= m\,(p^*_U - p^*_A p^*_B) + m(m-1)\left(p^*_{A\setminus B} p^*_{B\setminus A}(1-2/m)^{|I|} - p^*_A p^*_B\right) \\ &= m\left(e^{-t(c_A+c_B-c_I)} - e^{-t(c_A+c_B)} - c_I t e^{-t(c_A+c_B)}\right) \ \ ,\end{aligned}$$

where the second equality follows from several steps of elementary calculus that we omit here. By Eq. (5) we can then conclude

$$\lim \sigma(\hat{n}_A, \hat{n}_B) = m\left(e^{tc_I} - tc_I - 1\right) = \lim \mathbb{V}(\hat{n}_I)$$

Inserting this result in Eq. (4), and noting that for fixed $c_A \leq c_B$ it holds that $\lim \mathbb{V}(\hat{n}_A) \leq \lim \mathbb{V}(\hat{n}_B)$ (see eq. (2)), in particular $\mathrm{Var}[\widehat{n}_{S\cap S'}] \leq \mathrm{Var}[\widehat{n}_S]$ as well as $\mathrm{Var}[\widehat{n}_{S\cap S'}] \leq \mathrm{Var}[\widehat{n}_{S'}]$, yields

$$\begin{aligned}\lim \mathbb{V}(\widetilde{n}_{S,S'}) =& \lim(\mathbb{V}(\widehat{n}_{S\cup S'}) + 2\mathbb{V}(\widehat{n}_{S\cap S'}) - \mathbb{V}(\widehat{n}_S) - \mathbb{V}(\widehat{n}_{S'})) \\ \leq& \lim(\mathbb{V}(\widehat{n}_{S\cup S'}) + 2\mathbb{V}(\widehat{n}_{S\cap S'}) - \mathbb{V}(\widehat{n}_{S\cap S'}) - \mathbb{V}(\widehat{n}_{S\cap S'})) \\ =& \lim \mathbb{V}(\widehat{n}_{S\cup S'})\end{aligned}$$

□

When estimating the flow, the intersection of the readings of sensor S in a time interval T_b are intersected with the readings of sensor S' in consecutive time interval T_e. Let ΔT denote the time period between those two intervals. Then we denote the estimator for the flow between S and S' for this time period as $\widetilde{n}^{\Delta T}_{S,S'}$. This method can straight-forwardly be extended to paths. The flow on a path $S_1 \rightarrow S_2 \rightarrow S_3$ can be represented as $|\mathcal{R}^{T_1}_{S_1} \cap \mathcal{R}^{T_3}_{S_2} \cap \mathcal{R}^{T_3}_{S_3}|$. This quantity can again be estimated using the inclusion-exclusion formula.

$$\widetilde{n}^{\Delta T}_{S_1\,S_2,S_3} = \widehat{n}_{S_1} + \widehat{n}_{S_2} + \widehat{n}_{S_3} - \widehat{n}_{S_1\cup S_2} - \widehat{n}_{S_1\cup S_3} - \widehat{n}_{S_2\cup S_3} + \widehat{n}_{S_1\cup S_2\cup S_3}$$

The drawback of estimating the flow on paths is that the accuracy decreases drastically in the number of nodes on the path. In conclusion, we now have two major sources of high variance. A high loadfactor that is necessary to comply with high privacy requirements and a large number of intersections, required to monitor long paths. In the following a method for reducing the variance of the estimators is presented that improves the estimation of count distinct at a single sensor, as well as union and intersection estimation. Through this improvement, a higher loadfactor can be chosen to increase privacy while maintaining the same estimation accuracy. Furthermore, this improvement allows for monitoring the flow on longer paths with sufficient accuracy.

4.3 Improved Estimator

The improved estimator is based on the idea that the average of independent estimations of the same quantity is again an equally biased estimator with lower variance [7]. Hence, at each sensor, not one sketch is constructed, but r different sketches using r different and independent hash functions. This yields r different intermediate estimates, $\widehat{n}^1, ..., \widehat{n}^r$. The improved estimator is then defined as the mean of these intermediate estimates, i.e., $\widehat{\eta} = \frac{1}{r}\sum_{i=1}^{r}\widehat{n}^i$. The $\widehat{n}^1, ..., \widehat{n}^r$ are maximum likelihood estimators for count distinct and as such they are normally distributed and independent with common mean and variance [24], i.e., for all $i \in \{1, ..., r\}$ it holds that $\widehat{n}^i \sim \mathcal{N}\left(\mathbb{E}[\widehat{n}], \mathbb{V}(\widehat{n})\right)$. Thus, the improved estimator is normally distributed with $\widehat{\eta} \sim \mathcal{N}\left(\mathbb{E}[\widehat{n}], \frac{1}{r}\mathbb{V}(\widehat{n})\right)$. The improved estimator has the same expected value as the intermediate estimates, that is, it is asymptotically unbiased, whereas the variance of the improved estimator is reduced by a factor of $1/r$. Furthermore, because the intermediate estimators are normally distributed and asymptotically unbiased, the improved estimator based on their mean is not only again a maximum likelihood estimator for the count distinct, it is also the uniformly minimum variance unbiased estimator and the minimum risk invariant estimator [23].

However, in the pathological event that a sketch becomes full, i.e., $\mathbf{u}_n = 0$, the estimate for the count distinct based on this sketch is infinity. If only one of the r sketches runs full, the estimator fails. This drawback can be circumvented by using the median of the intermediate results instead of their mean. The median is very robust to outliers but has also weaker error guarantees, i.e., to guarantee an error not larger than ϵ with probability $1 - \delta$, the mean estimator requires $r \geq z_{1-\delta}\sqrt{\mathbb{V}(\widehat{n})}/\epsilon$, the median method requires $r \geq \log(1/\delta)/\epsilon^2$ [8] intermediate estimators. Consequently, for $\epsilon < 1$, the mean estimator requires less intermediate estimates to be as accurate as the median method.

5 Privacy Analysis

The main threat to privacy in the presented application scenarios is the so called linking attack, i.e., an attacker infiltrates or takes over the monitoring system and links this knowledge to background information in order to draw novel conclusions. For example, in a standard monitoring system that distributes the sensor readings, i.e., the device addresses, an attacker that knows the device address of a certain person as background knowledge, and furthermore infiltrates the monitoring system, is able to track this person throughout the monitored area.

Sketching prevents these linking attacks in two ways, obfuscation and k-anonymity. Obfuscation is accomplished by hashing the device address to sketch positions. Hence, before an attacker is able to re-identify a device, she has to infer the employed hash function. However, this very basic obfuscation technique can be vanquished using statistical analysis on sensor readings. The second anonymization technique is accomplished by the natural property of sketches to compress the address space, implicating collisions of addresses when mapped to sketch positions. Whereas these collisions entail a loss in accuracy, they create a form of anonymity, because an attacker can only infer upon a set of devices whose addresses are all mapped to that very same sketch position.

Formally, a monitoring system guarantees k-anonymity (see Sweeney [25]), if every monitored entity is indistinguishable from at least k other entities. Using linear count sketches with a loadfactor t results in t collisions per bucket on expectation, as implied by the uniformity property of the hash function, i.e.,

$$\forall i \in \{0, ..., m-1\} : \mathbb{E}\left[\left|\{a \in \mathcal{R}_S^T : h(a) = i\}\right|\right] = t \ .$$

Hence, the expected level of anonymity is t. We denote this form of anonymity **expected k-anonymity**, because the number of collisions is not deterministically guaranteed as required by regular k-anonymity. For a mathematical derivation of a similar probabilistic guarantee in the context of Bloom filters the reader is referred to Bianchi et al. [5].

The union of sensor readings is estimated by the binary or of the individual sketches. The binary or of a set of sketches has a loadfactor at least as high as the individual sketches themselves. Therefore, the level of expected k-anonymity is at least as high.

The intersection of sensor readings can contain far less device addresses than the individual readings. A sketch that is created on the readings of the intersection has thus a lower loadfactor. Even so, the intersection estimator presented in this paper is based on the estimators of the individual sketches and their union; the sketch of the intersection is not constructed at all. Therefore, the level of k-anonymity of this estimator for the intersection is again at least as high as the anonymity of the individual sketches.

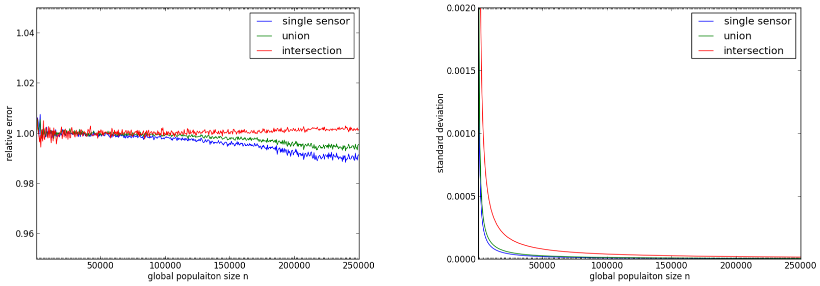
6 Experiments

In this section the empirical analysis of our method is presented. The general set up of experiments is as follows. A set of n addresses is randomly sampled out of a pre-defined address range ($\mathcal{A} = \{1, \cdots, 5 \times 10^7\}$). Out of this set we repeatedly sample with duplicates. The set is partitioned into k subsets $S_1, \cdots, S_k$ where S_i represents the sensor readings of sensor i. For each sensor S_i, a sketch sk_i of size m is generated using a global hash function h for all sensors. The estimate of sketches, their unions and intersections are then calculated as explained above.

6.1 Properties of the Estimator

For the first experiment, we simulate 3 sensors and vary the number of persons inside the sensor range from 500 to 250,000. Results for the average ratio of estimator and true value and the standard deviation of the ratio are shown in Fig. 2(a) and Fig. 2(b), respectively. The estimate is highly accurate—the error is always below 1%. Compared to the error introduced by the inference of the number of persons present in the area from the number of active Bluetooth devices [18], the error is negligible.

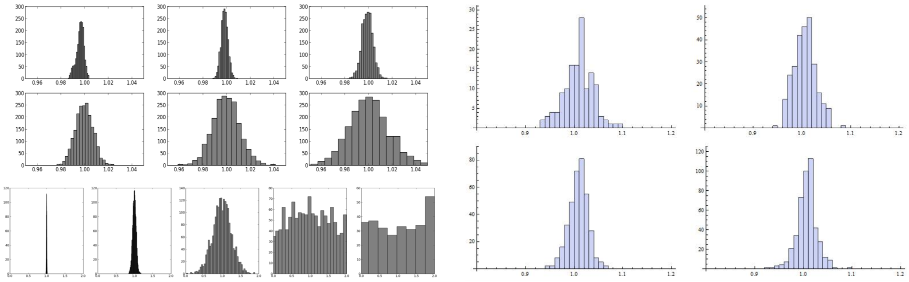
For simple Linear Counting, these results confirm existing expectations from theory and experimental studies. We need not go into a detailed comparison with other sketching methods in this paper, since two recent studies [19, 15] have done that already. One of the basic findings in these studies is that Linear Counting gives, using a suitable loadfactor, much more accurate estimates of the number of distinct objects than other sketching methods, e.g. FM-sketches or sampling based methods. This holds especially for small set sizes—where a number of 10,000 might already be considered small. For our application scenario this is important, since the size—especially of the intersections—can decrease to a few hundred persons. The experiment goes beyond the existing studies by showing that for the intersection of two sets the error can also be very low with a suitable loadfactor and that we can always set up a very accurate estimator using Linear Counting. A significant error of the estimator comes in only because we deliberately trade privacy against accuracy. As shown in section 5 the basic mechanism responsible for privacy is increasing the loadfactor. We analyze this trade-off, i.e., we investigate the impact of the loadfactor on the accuracy of estimates of one sensor as well as of intersections of up to five sensors. We simulate 5 sensors, and vary the loadfactor and the number of intersections. We average the results over 2,000 runs. The results are depicted in Fig. 3(a).



(a) (b)

Fig. 2: Average estimate $\widehat{n}$ (a) and average standard deviation (b) relative to the true value n for a loadfactor of 1.

The results confirm that the standard error increases with the loadfactor (Fig. 3(a), upper part), and even more rapidly with the number of intersections (Fig. 3(a), lower part). From this experiment we conclude that simple Linear Counting is indeed suitable for loadfactors smaller than 2 and intersection of at most two sensors. But for higher loadfactors or more intersections the trade-off can become unacceptable. This finding motivates the improved estimator investigated in the following.



(a) (b)

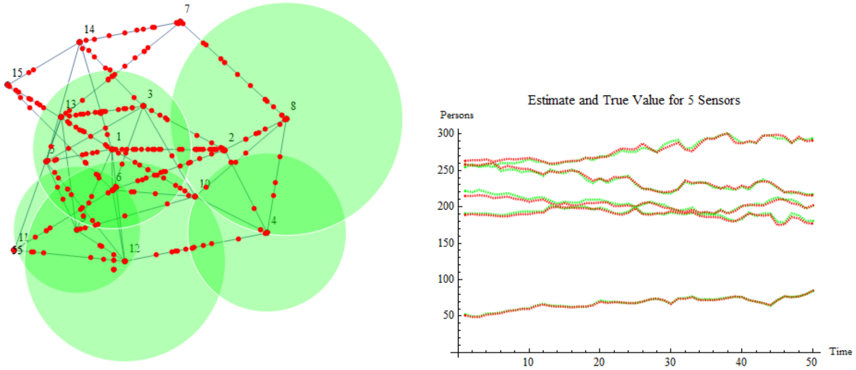
Fig. 3: (a) Distribution of estimates for loadfactors $0.5, 1, 2, 3, 4, 5$ (upper part) and numbers of sensors intersected, $1, 2, 3, 4, 5$ with a constant loadfactor of 5 (lower part). (b) Distribution of estimates using improved estimator with number of intermediate estimates r set to 5 (upper left), 15, 30 and 50 (lower right).

For that, we concentrate on a loadfactor of 5, because in regular k-anonymity 5 is a common value to ensure privacy. For each sensor, we take $r \in \{5, 15, 30, 50\}$ different random initializations of the hash function, resulting in r different sketches per sensor. Results, averaged over 500 runs, are shown in Fig. 3(b). The mean of estimates reduces the variance and is close to the true value. A good trade-off between the increase in accuracy and the higher memory requirements for storing multiple sketches is in the range between 15-30 sketches, since for higher numbers of sketches the variance reduction per additional sketch becomes insignificant. With these results we have demonstrated how to achieve a good trade-off between accuracy, privacy level, and memory consumption.

6.2 Real-World Simulation

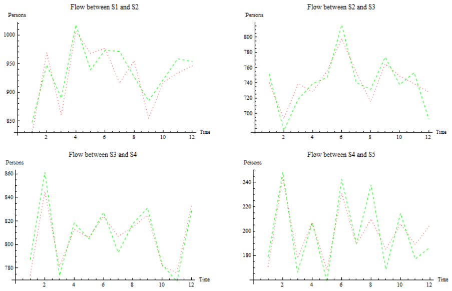
To investigate the flow and crowd monitoring in a more realistic setting, we implemented a simulation environment as follows. A random graph of k nodes with Bernoulli Graph Distribution ($p = 0.4$) is generated; the position of nodes in 2D-space is calculated using an automatic graph layout method. A number s of node locations is attached with sensors with a predefined range. In general, a sensor may cover more than one node and several edges. A number of n objects, i.e. the global population, is created. For each object a random sequence of tour stops (nodes of the graph) is generated. For every pair of tour stops the shortest path is determined using Dijkstra's method and inserted into the sequence between the stops. Finally, to each object a velocity, starting time and a step size is assigned (the latter because objects are not only at the node positions, but travel along the edges). During the simulation, for each time step the objects follow the tour with the assigned velocity and starting time, and their position along the edges is calculated. Each sensor monitors at each time step the objects in its sensor range. For each sensor and time period a new sketch is calculated and stored. As a ground truth, also the object address are stored. The simulation stops when the last object has completed its tour.

For the crowd monitoring scenario, overlapping sensors are simulated. Fig. 4(a) shows a snapshot from a simulation run. For the flow monitoring we use non-overlapping areas. The main difference compared to the experiments discussed in the last section is that the distribution of objects at nodes is not independent from each other because of flow constraints along the graph. The distribution is generated by a process very similar to real traffic flow, so that we have realistic flow properties over time. Fig. 4(b) shows an example for crowd monitoring with 15 nodes, 5 sensors, 1000 objects and a loadfactor of 5. Evidently, the estimates closely track the true values, as expected from the theoretical analysis and the experiments reported in the last section. For the flow monitoring scenario, Fig. 5(a) shows the estimate (green, dashed) and true (red, dotted) value of the flows between 4 sensors over time. Next we simulated the OD-matrix construction problem. Table 5(b) shows the relative error of an OD-matrix construction for 20,000 persons moving over a period of 15 time steps in a system



(a) (b)

Fig. 4: (a)Crowd simulation with Random Bernoulli Graph, 1000 persons (red dots), 5 Bluetooth sensors with overlapping range. (b) Estimated value (green dashed) and true value (red dotted) for each sensor for the first 50 time steps.



(a)

	S1	S2	S3	S4	S5
S1	0.005	0.008	0.007	0.018	0.001
S2	0.026	0.026	0.01	0.006	0.015
S3	0.004	0.003	0.016	0.002	0.014
S4	0.02	0.03	0.011	0.004	0.012
S5	0.014	0.005	0.004	0.017	0.027

(b)

Fig. 5: Flow estimation (green) and true values (red) between four sensors (a) as well as relative error of an OD-matrix between five sensors (b). Results are from flow simulation with 20000 persons and 5 sensors.

with 12 nodes, tracked by 5 sensors. Each sensor uses the improved estimator with 30 sketches and a loadfactor of 5. From these results we conclude that even for moderately sized sets, the error is very low. Overall, we conclude from the experiments that Linear Counting behaves in the more complex setting of the simulation as expected from a theoretical point of view and is a very promising approach for deployment in real-world applications.

7 Discussion

In this paper we present a new, privacy aware method for estimating flows and tracks as well as for estimating OD-matrices. This way, we extended the Linear Counting approach to a general set of primitives for privacy-preserving mobility modeling. We show theoretically and empirically that two challenging application scenarios can be solved using and combining this set of primitives. To compensate the accuracy deficit of Linear Counting for strict privacy requirements we present a method for increasing the accuracy, while maintaining the privacy. This method is also applicable to boost accuracy in flow estimation, allowing to monitor even tracks.

In contrast to many privacy-preserving approaches, this one is easy to implement, has excellent accuracy and can be implemented efficiently. Our experiments suggest that it is immensely useful in a practical settings and can have a real impact on how stationary sensor based data collection is done.

While being accurate on count distinct and flow estimation, even for strict privacy, the accuracy of our method drops drastically with the length of monitored tracks. The improved estimator can compensate this drop to a certain level. However, experiments show that estimating tracks of length greater 5 leads to large errors. Therefore, we recommend using our method on count distinct and flows. When monitoring tracks, depending on their length, a user might have to reduce privacy requirements in order to maintain a certain accuracy.

The main drawback of Linear Counting when compared to other sketching techniques is the memory usage. Most sketching techniques, e.g., FM Sketches, use memory logarithmic in the number of items it estimates. The linear count sketches, however, have linear memory usage, leading to potentially large sketches. Fortunately, stationary sensors usually can be equipped with large memory (e.g., 32GB flash memory). Hence, this is unproblematic for our application scenarios. Still, the memory footprint can become problematic, because communication is in general costly. If large sketches have to be send very frequently, communication costs can become significant, or sketch sizes might even exceed network capacities.

In follow up research, we want to tackle the general problem of communication costs when using stationary sensors. However, when monitoring non-linear functions, like the union or intersection of sets, this task is not trivial. The LIFT-approach provides a framework for communication reduction in distributed systems, allowing communication efficient monitoring of non-linear functions. We want to apply the LIFT-approach to our monitoring system and test the benefits of employing the LIFT-approach in a real-world experiment.

8 Acknowledgements

This research has been supported by the EU FP7/2007-2013 under grant 255951 (LIFT) and by the German Science Foundation under ‘GA 1615/2-1’.

References

[1] C.C. Aggarwal and P.S. Yu. A general survey of privacy-preserving data mining models and algorithms. *Privacy-preserving data mining*, pages 11–52, 2008.

[2] K. Ashok and M.E. Ben-Akiva. Dynamic origin-destination matrix estimation and prediction for real-time traffic management systems. In *International Symposium on the Theory of Traffic Flow and Transportation*, 1993.

[3] J. Barceló, L. Montero, L. Marquès, and C. Carmona. Travel time forecasting and dynamic origin-destination estimation for freeways based on bluetooth traffic monitoring. *Transportation Research Record: Journal of the Transportation Research Board*, 2175(1):19–27, 2010.

[4] L. Bergamini, L. Becchetti, and A. Vitaletti. Privacy-preserving environment monitoring in networks of mobile devices. In *NETWORKING 2011 Workshops*, pages 179–191. Springer, 2011.

[5] Giuseppe Bianchi, Lorenzo Bracciale, and Pierpaolo Loreti. better than nothing privacy with bloom filters: To what extent? In *Privacy in Statistical Databases*, pages 348–363. Springer, 2012.

[6] Andrei Broder and Michael Mitzenmacher. Network applications of bloom filters: A survey. *Internet Mathematics*, 1(4):485–509, 2004.

[7] G. Cormode and M. Garofalakis. Sketching probabilistic data streams. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pages 281–292. ACM, 2007.

[8] G. Cormode, M. Datar, P. Indyk, and S. Muthukrishnan. Comparing data streams using hamming norms (how to zero in). *Knowledge and Data Engineering, IEEE Transactions on*, 15 (3):529–540, 2003.

[9] G. Cormode, S. Muthukrishnan, and K. Yi. Algorithms for distributed functional monitoring. *ACM Transactions on Algorithms (TALG)*, 7(2):21, 2011.

[10] A.C. Davies, J.H. Yin, and S.A. Velastin. Crowd monitoring using image processing. *Electronics & Communication Engineering Journal*, 7(1):37–47, 1995.

[11] C. Dwork. Differential privacy. *Automata, languages and programming*, pages 1–12, 2006.

[12] Arik Friedman, Ran Wolff, and Assaf Schuster. Providing k-anonymity in data mining. *The VLDB Journal*, 17(4):789–804, 2008.

[13] S. Ganguly, M. Garofalakis, and R. Rastogi. Tracking set-expression cardinalities over continuous update streams. *The VLDB Journal*, 13(4):354–369, 2004.

[14] P.B. Gibbons. Distinct-values estimation over data streams. In *In Data Stream Management: Processing High-Speed Data*. Springer, 2009.

[15] N. Gonçalves, R. José, and C. Baquero. Privacy preserving gate counting with collaborative bluetooth scanners. In *On the Move to Meaningful Internet Systems: OTM 2011 Workshops*, pages 534–543. Springer, 2011.

[16] J. Heikkila and O. Silvén. A real-time system for monitoring of cyclists and pedestrians. In *Second IEEE Workshop on Visual Surveillance (VS'99)*, pages 74–81. IEEE, 1999.

[17] M. Langheinrich. Privacy by design - principles of privacy-aware ubiquitous systems. In *Ubicomp 2001: Ubiquitous Computing*, pages 273–291. Springer, 2001.

[18] T. Liebig, Z. Xu, M. May, and S. Wrobel. Pedestrian quantity estimation with trajectory patterns. In *Proceedings of the ECML/PKDD*, 2012.

[19] Ahmed Metwally, Divyakant Agrawal, and Amr El Abbadi. Why go logarithmic if we can go linear?: Towards effective distinct counting of search traffic. In *Proceedings of the 11th international conference on Extending database technology: Advances in database technology EDBT*, pages 618–629. ACM, 2008.

[20] D. Mir, S. Muthukrishnan, A. Nikolov, and R.N. Wright. Pan-private algorithms via statistics on sketches. In *Proceedings of the 30th symposium on Principles of database systems of data*, pages 37–48. ACM, 2011.

[21] A. Monreale. *Privacy by Design in Data Mining*. PhD thesis, University Pisa, 2011.

[22] Bart Preneel. *Analysis and design of cryptographic hash functions*. PhD thesis, Katholieke Universiteit te Leuven, 1993.

[23] F. Rusu and A. Dobra. Statistical analysis of sketch estimators. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pages 187–198. ACM, 2007.

[24] S.G. Self and K.Y. Liang. Asymptotic properties of maximum likelihood estimators and likelihood ratio tests under nonstandard conditions. *Journal of the American Statistical Association*, 82(398):605–610, 1987.

[25] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10.

[26] K.Y. Whang, B.T. Vander-Zanden, and H.M. Taylor. A linear-time probabilistic counting algorithm for database applications. *ACM Transactions on Database Systems (TODS)*, 15 (2):208–229, 1990.