Research Article

Achieving the Optimal k -Anonymity for Content Privacy in Interactive Cyberphysical Systems

Jinbao Wang ¹, Ling Tian ², Yan Huang,³ Donghua Yang ¹ and Hong Gao⁴

¹The Academy of Fundamental and Interdisciplinary Sciences, Harbin Institute of Technology, China

²School of Computer Science and Technology, University of Electronic Science and Technology of China, China

³Department of Computer Science, Georgia State University, USA

⁴School of Computer Science and Technology, Harbin Institute of Technology, China

Correspondence should be addressed to Ling Tian; lingtian@uestc.edu.cn and Donghua Yang; yang.dh@hit.edu.cn

Received 14 June 2018; Revised 18 August 2018; Accepted 29 August 2018; Published 26 September 2018

Academic Editor: Liran Ma

Copyright © 2018 Jinbao Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Modern applications and services leveraged by interactive cyberphysical systems (CPS) are providing significant convenience to our daily life in various aspects at present. Clients submit their requests including query contents to CPS servers to enjoy diverse services such as health care, automatic driving, and location-based services. However, privacy concerns arise at the same time. Content privacy is recognized and a lot of efforts have been made in the literature of privacy preserving in interactive cyberphysical systems such as location-based services. Nevertheless, neither the cloaking based solutions nor existing client based solutions have achieved effective content privacy by optimizing proper content privacy metrics. In this paper we formulate the problem of achieving the optimal content privacy in interactive cyberphysical systems using k -anonymity solutions based on two content privacy metrics, which are defined using the concepts of entropy and differential privacy. Then we propose an algorithm, Multilayer Alignment (MLA), to establish k -anonymity mechanisms for preserving content privacy in interactive cyberphysical systems. Our proposed MLA is theoretically proved to achieve the optimal content privacy in terms of both the entropy based and the differential privacy mannered content privacy metrics. Evaluation based on real-life datasets is conducted, and the evaluation results validate the effectiveness of our proposed algorithm.

1. Introduction

Cyberphysical systems (CPS), which deeply integrate different computing, communication, controlling, and monitoring components, have leveraged modern services in our daily life, like smart grid, intelligent transportation, automatic driving, etc. Recent development of mobile communication and networks has leveraged many modern applications built on interactive cyberphysical systems, in which client software programs or devices take actions according to their interactions with CPS servers. In more details, a client sends a request to the CPS server and is to take actions on receiving the reply from the CPS server. The actions to be taken depend on the reply of CPS servers. Health caring, automatic driving, and location-based services fall into this category of interactive CPS applications. Suppose an old guy Bob is wearing a health caring device, which is connected

to a CPS server though mobile Internet. Bob could send “stomachache” to the CPS server for instructions to help him. The CPS server may send a reply telling Bob what to do or where is the nearest hospital. Then Bob takes his action according to the reply. Similar processes could be adopted in applications of automatic driving and location-based services. These modern services over interactive CPS are attractive since they do bring convenience to people’s daily life. However, privacy concerns arise at the same time while users’ requests are submitted to the CPS servers through Internet. These requests disclose users’ query contents to the CPS server and even vicious listeners to the communication channels. Here we refer query content as parameters in users’ requests, such as “stomachache” in Bob’s request to the health caring CPS server. These query contents should be kept as sensitive information for individuals, and the abuse or further leakage of these information will make users

vulnerable in respect of private life or even individual security [1].

To the privacy concern, content privacy should be recognized to emphasize that users' query contents should be kept as sensitive information in interactive cyberphysical systems. Many research efforts have been made to protect different types of query contents such as locations and keywords in the literature of interactive cyberphysical systems such as location-based services. The major body of these efforts consists of two parts, cloaking based solutions and client based solutions. Cloaking based solutions employ a trusted server from a third party. When a user u queries with a query content $u.content$, the query $q(u.content)$ is sent to the trusted server which in the next step generates a cloaking region CR containing u 's location and at least another $k - 1$ users. Here k specifies the level of privacy guarantee. Then the query with CR is sent to the CPS server, and the CPS server could not determine where u is or even whether u is querying. In this process, cloaking based solutions aim to make u indistinguishable from another $k - 1$ users; however it suffers from inherent drawbacks brought by the trusted server, which may become the single point of failure of privacy and the bottleneck of query performance. More seriously, when the CPS server holds certain side information such as the prior probability of query contents, cloaking based solutions will suffer from further privacy breach. To address this issue, client based solutions are presented, aiming at k -anonymity provided by the client side. Reference [2] generates $k - 1$ dummy query contents for continuous scenarios. To prevent the adversary from inferring the actual query content, [2] constrains that the selected dummies should have prior probability larger than a predefined threshold. In practice, it is difficult to determine a proper threshold, and what is more, dummies selected by [2] could still be eliminated if they have quite different prior probability. Reference [3] employs an entropy based privacy metric and generates $k - 1$ dummy locations in random manner. However, the improper dummies could be eliminated from the reported k query contents due to the process of [3]; thus the provided privacy is degraded.

In this paper, we investigate the problem of preserving content privacy in interactive cyberphysical systems with a client based solution. To guarantee the utility of requests to the CPS servers, we adopt k -anonymity in order to prevent the adversary from recognizing the actual query content from the k reported contents, since the actual query content must be sent to the CPS server for a meaningful reply. In this process, the major challenge arises in two aspects. First, the k -anonymity provided should be carefully designed so that the adversary is not able to eliminate any query contents. Second, the overall privacy produced by k -anonymity should be optimized. We present two privacy metrics denoted expected entropy and dp-ratio which depict the achieved content privacy using a entropy based concept and a differential privacy mannered measurement, respectively. Then an algorithm, Multilayer Alignment (MLA), which establishes k -anonymity based mechanisms for preserving content privacy is proposed. Given the prior probability of

query contents together with an integer k which specifies the privacy level, MLA generates a set of reports, each of which consists of k distinct query contents, together with probability distribution on the report set for each query content. Given any report r submitted to the CPS server, MLA guarantees that the posterior probability of each query content in r is larger than 0. To this end, the adversary is not able to eliminate any query content from a report. Here a report could be taken as a set of different query contents, and we give its formal definition in Section 3. We theoretically introduce the properties of MLA by proving that MLA achieves the optimal expected entropy and the optimal dp-ratio at the same time. These attractive properties make MLA the optimal k -anonymity solution for preserving content privacy in interactive cyberphysical systems. The major contributions of this paper are as follows.

- (i) We formulate the problem of achieving the optimal k -anonymity based mechanisms for preserving content privacy in interactive cyberphysical systems. The problem formulation is based on two content privacy metrics with entropy and differential privacy concepts.
- (ii) We propose the Multilayer Alignment (MLA) algorithm, which establishes k -anonymity based mechanisms for preserving content privacy. The MLA algorithm prevents adversaries from eliminating query contents from reports using Bayes inference.
- (iii) We prove that MLA achieves the optimal k -anonymity mechanisms in terms of our presented content privacy metrics simultaneously.
- (iv) We evaluate our proposed MLA algorithm using real-life datasets. The evaluation results validate that MLA achieves effective content privacy in terms of the entropy based and differential privacy mannered content privacy metrics.

The rest of this paper is organized as follows. Section 2 introduces some necessary preliminaries including the process of preserving content privacy using a client based solution, together with common accepted privacy metrics. Section 3 formulates the problem of achieving the optimal k -anonymity for content privacy in interactive cyberphysical systems. Section 4 proposes the MLA algorithm which establishes effective mechanisms for preserving content privacy. Section 5 theoretically proves that our proposed MLA algorithm achieves the optimal k -anonymity for content privacy in terms of the content privacy metrics introduced in Section 3. Section 6 evaluates the MLA algorithm based on real-life datasets and related work of this paper is discussed in Section 7. Finally, Section 8 concludes this paper.

2. Preliminary

This section introduces necessary preliminaries including the process of preserving content privacy using a client based k -anonymity solution. Then two accepted privacy

notions, i.e., k -anonymity and differential privacy, and their corresponding metrics are introduced.

2.1. Client Based k -Anonymity Solution. In the process of a client based k -anonymity solution for content privacy preservation, k query contents are reported to the CPS server when a user u wants to submit a request. The k reported query contents are determined on u 's device, and no trusted third-party servers are employed; thus the potential single point of privacy failure and query performance bottleneck are eliminated. It is worth noticing that the actual query content u queries should be included in the reported ones; otherwise u is not able to receive a meaningful reply from the CPS server. After receiving the reported query contents, the CPS server processes k queries, one for each reported query content, and then returns the query results to u . Irrelevant results are filtered on u 's mobile device and the actual results are returned to u . In this process, the CPS server receives k distinct query contents instead of a single actual one, and to this end the actual query content is hidden. Nevertheless, careful design is required to avoid ineffective dummies in the reported query contents. The way of generating reports to the CPS server determines the level of content privacy achieved, and this motivates our work.

2.2. Privacy Notions and Metrics

2.2.1. k -Anonymity. One widely adopted notion of privacy is k -anonymity, which is firstly introduced in the database community by [4]. The principle of k -anonymity is to hide the sensitive information into $k - 1$ dummies so as to make the adversary unable to recognize the actual one. In the literature of privacy protection in interactive cyberphysical systems, k -anonymity could be categorized into cloaking based solutions and client based solutions. The cloaking based solutions such as [5] employ a third-party but trusted server, which is responsible for hiding the actual user among at least $k - 1$ dummy users by spatial generalization. The client based solutions including more recent work such as [2, 3, 6] perform on users' devices and generate $k - 1$ dummies in a local manner, in which process certain side information is adopted, for instance, the prior probability of each query content. The trusted server in cloaking based solutions may become a single failure if it is hacked by attackers and it is the performance bottleneck to incur long latency to requests. What is more, most of cloaking based solutions is unaware of side information held by the adversary such as the prior probability of query contents. At the same time, the existing client based solutions provide specious k -anonymity, since the attackers may violate the principle of k -anonymity through rerunning of the algorithms or launching probability inference for each of the k query contents.

The quality of k -anonymity could be measured by the concept of entropy borrowed from the area of information theory. When the CPS server receives a report $r = \{c_1, \dots, c_k\}$ consisting of k query contents, the entropy of r is formulated as follows:

$$H(r) = -\sum_{i=1}^k Pr(c_i | r) \log Pr(c_i | r) \quad (1)$$

Note that the former formulation is slightly different from [3]. Actually, [3] takes the prior probability $Pr(c_i)$ as an approximation of the posterior probability $Pr(c_i | r)$.

2.2.2. Differential Privacy. Differential privacy was firstly introduced and applied in statistic databases, and it aims to prevent the leakage of any individual's information during query processing. Generally speaking, to satisfy the notion of differential privacy, a random algorithm should return query results with similarly distribution for two databases differing with just one tuple. In other words, a single modification in a database brings a minor change to query results under the control of differential privacy. The definition of differential privacy is given below.

Definition 1 (differential privacy). Given $\epsilon \geq 0$, a randomized algorithm $\mathcal{A}$ satisfies ϵ -differential privacy if for all neighboring databases D and D' , $Prob(\mathcal{A}(D) \in S) \leq e^\epsilon \times Prob(\mathcal{A}(D') \in S)$. Here $S \subseteq Range(\mathcal{A})$. Any pair of neighboring databases D and D' satisfies one of the following conditions: (1) (for unbounded differential privacy) D can be transformed to D' with exact one insertion or deletion; (2) (for bounded differential privacy) D can be transformed to D' with exact one modification.

The bounded differential privacy prevents distinguishing two datasets with the same size while differing with exact one tuple. The unbounded differential privacy prevents distinguishing two datasets which are the same except that one of them holds exact one additional tuple.

The metric for differential privacy is the coefficient ϵ in Definition 1. Intuitively, a smaller ϵ leads to a better privacy but larger noise in the query result, while a larger ϵ leads to less noise in the query result but a weaker privacy guarantee.

3. Problem Definition

This section formulates the problem of achieving the optimal k -anonymity based mechanisms for content privacy in interactive cyberphysical systems. Before introducing the problem definition, we provide several definitions which interpret indispensable concepts for our problem definition.

When a user queries a content c , the client based solution first generates a report consisting of k distinct query contents and then sends the report to the CPS server for response. A formal definition of a report is given as follows.

Definition 2 (report). Given the global set C of query contents and an integer $k > 0$, a report r is a subset of C with size k . When a user queries content c , the generated report r must contain c ; i.e., $c \in r$. Denote the set of all the reports for the given C and k by R_C^k . The set of reports containing the query content c is denoted by $R_C^{C,k}$.

In the rest of this paper, we focus on specified C and k , and we also use the notion R_c (instead of $R_C^{C,k}$) to refer to the set of reports containing the query content c .

For a client based solution, multiple reports could include an identical query content c . When c is queried, one of these reports is submitted to the CPS server. The following definition of reporting probability depicts the process of selecting such a report.

Definition 3 (reporting probability). Given the global set C of query contents and an integer $k > 0$, a reporting probability is a function $Pr_C^k : C \times R_C^k \rightarrow [0, 1]$ satisfying the following constraints:

- (i) for any $c \in C$ and $r \in R_c$, $Pr_C^k(c, r) \geq 0$;
- (ii) for any $c \in C$, $\sum_{r \in R_c} Pr_C^k(c, r) = 1$;
- (iii) for any $c \in C$ and $r \notin R_c$, $Pr_C^k(c, r) = 0$.

The first two constraints in the definition of reporting probability illustrate that when querying a content $c \in C$, a report $r \in R_c$ is selected according to the probability $Pr_C^k(c, r)$. The third constraint specifies that a report r will not be selected for c if $c \notin r$.

Next we formulate a client based solution as a mechanism in a probabilistic manner based on the concepts of report and reporting probability.

Definition 4 (mechanism). Given the global set C of the query contents and an integer $k > 0$, a k -anonymity based mechanism $\mathcal{M}$ consists of two components including the set of reports R_C^k and the reporting probability Pr_C^k . When a user queries content $c \in C$, $\mathcal{M}$ randomly selects a report r from R_C^k and the probability of selecting r is $Pr_C^k(c, r)$.

The above definition of a k -anonymity based mechanism looks speciously strange; nevertheless existing solutions could be taken as instances of the above definition. We could specify the reporting probability using additional parameters in these works, for instance, the predefined prior probability threshold σ in [2].

This paper adopts two privacy metrics to measure a given k -anonymity based mechanism $\mathcal{M}$ in terms of privacy. As formulated in the following definition, the first metric integrates entropy measures of all the reports generated by $\mathcal{M}$.

Definition 5 (expected entropy). Given the global set C of query contents, an integer $k > 0$ and the prior probability of query contents as $pr(\cdot)$, the expected entropy of mechanism $\mathcal{M}(R_C^k, Pr_C^k)$ is calculated as

$$\begin{aligned} H_E(\mathcal{M}) &= \sum_{r \in R_C^k} Pr(r) H(r) \\ &= \sum_{r \in R_C^k} \sum_{c \in r} pr(c) Pr_C^k(c, r) \sum_{c \in r} -Pr(c | r) \log Pr(c | r). \end{aligned} \quad (2)$$

Here $Pr(c | r) = pr(c)Pr_C^k(c, r) / \sum_{i \in r} pr(i)Pr_C^k(i, r)$, and it is the posterior probability of c given report r . $H_E(\mathcal{M})$ measures the achieved content privacy overall by considering

all the generated reports. The probability of each report is taken as the weight, and the entropy of each report is integrated in the above formulation. A larger $H_E(\mathcal{M})$ indicates that a better content privacy is obtained with respect to the concept of entropy.

The second metric incorporates the notion of k -anonymity and differential privacy. It measures a mechanism with the most distinguishable pairs of query contents in the generated reports. The following definition formulates our second metric named dp-coefficient.

Definition 6 (dp-coefficient). Given the global set C of query contents, an integer $k > 0$, and the prior probability of query contents as $pr(\cdot)$, the dp-coefficient of mechanism $\mathcal{M}(R_C^k, Pr_C^k)$ is calculated as

$$dpc(\mathcal{M}) = \max_{r \in R_C^k, i, j \in r} \ln \frac{Pr(i | r)}{Pr(j | r)}. \quad (3)$$

Here the terms $Pr(i | r)$ and $Pr(j | r)$ are the posterior probability of query contents i and j given a report r , and they could be calculated in the same way as the calculation of $Pr(c | r)$ described above.

Based on the content privacy metrics, e.g., expected entropy and differential privacy coefficient, we formulate the problem of achieving the optimal k -anonymity for content privacy in interactive cyberphysical systems as follows.

Problem Definition. Given the global set C of query contents, an integer $k > 0$, and the prior probability of query contents as $pr(\cdot)$, compute a mechanism $\mathcal{M}(R_C^k, Pr_C^k)$ with the optimal content privacy. The optimal content privacy is achieved if $H_E(\mathcal{M})$ is maximized.

$H_E(\mathcal{M})$ and $dpc(\mathcal{M})$ depict the content privacy achieved by $\mathcal{M}$ from the holistic and individual point of view, respectively. Although our problem definition aims at the optimized expected entropy, in the next section we propose an algorithm which achieves the optimal expected entropy and the optimal dp-coefficient simultaneously.

4. Achieving the Optimal k -Anonymity

This section in first provides a short discuss on a naïve solution to the problem defined in Section 3. Then we propose our Multilayer Alignment (MLA) algorithm which achieves the optimal k -anonymity for content privacy in interactive cyberphysical systems. MLA exhibits an attractive property that it achieves the maximized expected entropy and the minimized dp-coefficient simultaneously.

4.1. A Naïve Solution. According to the problem definition in Section 3, the essential challenge of establishing the optimal mechanisms lies in building the reporting probability $Pr_C^k : C \times R_C^k \rightarrow [0, 1]$. A naïve approach to achieve the optimal k -anonymity is formulating the problem using nonlinear-programming technique with linear constraints in Definition 3, and expected entropy or dp-coefficient is used as the

optimizing objective. However, the nonlinear-programming formulation employs $\binom{|C|}{k} \times |C|$ variables each of which stands for an entry in Pr_C^k . When $|C|$ grows to 100 and k is set to 10, there will be more than 10^{15} variables. Thus this naïve approach is impractical due to its computation expense.

4.2. The Multilayer Alignment Algorithm. MLA computes the optimal k -anonymity mechanism in two phases, namely, (1) Segment Alignment and (2) Mechanism Initiation. The major idea of MLA is to generate a mechanism where query contents have as similar posterior probability as possible in each report. To accomplish this goal in a holistic manner, Segment Alignment amortizes each query content with large prior probability to multiple query contents with small prior probability. What is more, the reports generated by Mechanism Initiation have the same distribution of posterior probability of the k included query contents. Next we introduce the two phases of MLA.

4.2.1. Segment Alignment. Given the prior probability $pr(\cdot)$ of query contents, MLA represents each query content $c_i \in C$ using a segment s_i with length $|s_i| = pr(c_i)$. The segments for all the query contents are sorted in descend order, and denote the sorted set as $Seg = \{s_1, \dots, s_m\}$. Then MLA aligns the segments onto k layers in order. The aligning process has two modes, i.e., aligning dominant and aligning dominated. At the beginning of aligning, the mode of aligning dominant is active. The number of rest layers (denoted $rest_{layer}$) is set to k . MLA checks whether the current segment is dominant. When aligning s_i , s_i is dominant if the condition $|s_i| \times rest_{layer} > \sum_{i \geq j \geq m} |s_j|$ holds. If the current segment s_i is dominant, MLA aligns s_i onto the current layer and s_i takes up the entire layer. The aligning stays in mode aligning dominant, and segment s_{i+1} is taken as the current segment when the aligning continues. If the current segment s_i is not dominant, the aligning turns to mode aligning dominated. Then MLA sets the length of each of the remaining layers as $\sum_{i \geq j \geq m} |s_j| / rest_{layer}$, and it aligns $s_i, \dots, s_m$ along the rest layers. When aligning a segment s_j and the current layer has blank length l_b less than $|s_j|$, s_j is divided into two parts with lengths l_b and $|s_j| - l_b$. The first part is aligned onto the current layer, and the second part is aligned onto the beginning of the next layer. In the mode of aligning dominated, MLA goes on aligning all the remaining segments, and it never turns back to mode aligning dominant. After all the remaining segments are aligned, the first phase of MLA terminates and MLA continues to the second phase.

Example 7. Suppose $C = \{c_1, c_2, c_3, c_4, c_5, c_6, c_7\}$ and Alice wants to query c_1 nearby, and the prior probability of each query content at her location is given as follows: $pr(c_1) = 0.4$, $pr(c_2) = 0.15$, $pr(c_3) = 0.15$, $pr(c_4) = 0.1$, $pr(c_5) = 0.1$, $pr(c_6) = 0.05$, and $pr(c_7) = 0.05$. Alice desires for 4-anonymity ($k = 4$), and what is the optimal mechanism for Alice?

Here we use the instance in Example 7 to illustrate the process of segment aligning. There are 4 layers in the process

FIGURE 1: Segment aligning for the instance consisting of 7 query contents in Example 7. The first layer is taken up by c_1 under aligning dominant mode, and the remaining 3 layers are taken up by c_2, c_3, c_4, c_5, c_6 , and c_7 , under aligning dominated mode. Here c_{ij} means the j th part of content i .

of aligning. The segment for c_1 is aligned in mode aligning dominant since $pr(c_1) \times k > 1$. The segment for the remaining query contents is aligned in mode aligning dominated, and each of the remaining layers is at length $(1 - 0.4)/(4 - 1) = 0.2$. The aligning result is shown in Figure 1.

4.2.2. Mechanism Initiation. Denote the length of the i th layer by L_i , where $i = 1, \dots, k$. The second phase of MLA first shrinks the length of each layer, and the i th layer is shrunk by ratio L_i/L_k . The shrinking ratio of the i th layer is recorded as $ratio_i$. All the layers have the same length after shrinking. In the next step, MLA sets a vertical line at the beginning of each layer. Then the vertical line moves to the right until it touches the first point on any layer at which a segment ends. Then the scanned parts of the k layers are packed into a report. Denote the scanned part on the i th layer with length l_i^s ; then the probability of this report is $\sum_{i \geq j \geq k} l_i^s \times ratio_i$ and the posterior probability of the query content on the i th layer is $(l_i^s \times ratio_i) / (\sum_{i \geq j \geq k} l_j^s \times ratio_j)$. The vertical line continues moving to the right and MLA packs the next report when any segment ends on a layer. The process terminates after the vertical line moves to the end of each layer and generates the last report. Continue with Example 7 as shown in Figure 2, the shrinking ratios of the 4 layers are 2, 1, 1 and 1. Then a vertical line starts moving to the right from the left end of all the layers. It first touches the end points of c_{32} and c_5 on layers 3 and 4, respectively, and a report $r_1 = \{c_1, c_2, c_3, c_5\}$ is generated. Then it keeps moving to the right and touches the end points of c_2 and c_6 on layers 2 and 4, respectively, and report $r_2 = \{c_1, c_2, c_4, c_6\}$ is generated. Finally, the vertical line touches the end points of all the layers and generates the last report $r_3 = \{c_1, c_3, c_4, c_7\}$. In the end, MLA generates 3 reports including $r_1 = \{c_1, c_2, c_3, c_5\}$, $r_2 = \{c_1, c_2, c_4, c_6\}$, and $r_3 = \{c_1, c_3, c_4, c_7\}$. The reporting probability is given in Table 1. Take c_1 , for instance; half of its prior probability is assigned into report r_1 , and one-fourth of its prior probability is assigned to reports r_2 and r_3 . Thus the reporting probability of c_1 for reports r_1, r_2 , and r_3 is $1/2, 1/4$, and $1/4$, respectively, as shown in Table 1. The reporting probability of other contents could be calculated in the same manner.

FIGURE 2: The first layer is shrunk by ratio of 2, and remaining layers keep unchanged. Mechanism initiation produces three reports $r_1 = \{c_1, c_2, c_3, c_5\}$, $r_2 = \{c_1, c_2, c_4, c_6\}$, and $r_3 = \{c_1, c_3, c_4, c_7\}$. When querying c_1 , the mechanism adopts reporting distribution $Pr(r_1 | c_1) = 0.5$, $Pr(r_2 | c_1) = 0.25$ and $Pr(r_3 | c_1) = 0.25$.

TABLE 1: The reporting probability for Example 7.

Content / Report	r_1	r_2	r_3
c_1	1/2	1/4	1/4
c_2	2/3	1/3	0
c_3	2/3	0	1/3
c_4	0	1/2	1/2
c_5	1	0	0
c_6	0	1	0
c_7	0	0	1

The pseudocode of the Multilayer Alignment algorithm is shown in Algorithm 1. In the beginning, MLA sorts the query contents in C according to their prior probability in descending order (line 1). Then it initiates necessary structures and variables. $L_1, \dots, L_k$ stores the alignment of k layers (lines 2-3). Variable l_{cursor} indicates the layer being processed (line 4), and $isDominant$ indicates whether the alignment is under aligning dominant mode (line 5). Variable $partial$ indicates how much prior probability of the current query content is taken up by the last layer while $average$ indicates the length of each layer processed in dominated mode, and they are initiated in line 6. Array Len keeping the lengths of k layers is initiated in line 7. The loop in lines 8-28 aligns query contents in C in order onto k layers. The aligning mode is set dominant in line 5 before processing the first query content. Under aligning dominant mode, MLA checks whether c_i should be aligned under aligning dominant mode. If the answer is yes, a segment for c_i is created with length $pr(c_i)$ and it is added to the list $L_{l_{cursor}}$ for the current layer. Here the constructor of segment specifies the label and the length for a segment. Then the alignment of c_i and the current layer terminates (lines 10-12). If c_i should not be aligned under aligning dominant mode, MLA turns to the mode of aligning dominated and calculates the length of each of the remaining layers as $average$ (lines 13-15). In aligning dominated mode, MLA executes the code in lines 16-28. If c_i could be entirely aligned onto the current layer (line 16), MLA creates a segment for c_i with length $pr(c_i)$, and adds it to list $L_{l_{cursor}}$ (line 17). The length of the current layer and

partial are updated (lines 18-19). If the current layer does not have sufficient space to hold c_i , c_i is split into two segments. Lines 21-23 align the first segment onto the current layer, and lines 26-28 align the second segment onto the next layer. Lines 24-25 deal with a special case where c_i exactly uses up the space of the current layer. By here the Segment Alignment terminates and the MLA goes to the phase of Mechanism Initiation. It packs the heads of k lists L_i into a report r (lines 30-31). Then MLA determines the movement length ratio of the vertical line to the right as *step* (line 32). The reporting probability related to the current report is calculated in line 35. For each layer, MLA updates the length of the head. If the head of a layer is entirely packed into a report, it is popped from the list (lines 36-39).

The computation cost of MLA consists of three parts, sorting C and initiating variables, Segment Alignment, and Mechanism Initiation. The first part costs $O(|C| \log |C|)$. Segment Alignment costs $O(k + |C|)$ since at most $k + |C|$ segments are aligned, and each alignment costs a constant time. Mechanism Initiation costs $O(k(k + |C|))$ which is dominated by packing at most $k + |C|$ reports (each packing costs $O(k)$) and calculating at most $k(k + |C|)$ entries for Pr_C^k . In practice, k should be set smaller than $|C|$. The total cost of MLA is $O(|C| \log |C| + k|C|)$.

5. Properties of MLA Algorithm

This section formally proves that MLA achieves the optimal expected entropy and the optimal dp-coefficient simultaneously. We first introduce some concepts which build necessary foundation for our formal proof.

Definition 8 (dominant content). Given the global set C of query contents, an integer $k > 0$, and the prior probability $pr(\cdot)$, $\forall c \in C$, let $dom(c)$ be the number of query contents larger than c . A query content $c \in C$ is a dominant content iff the following conditions hold:

- (i) $dom(c) < k$;
- (ii) $\sum_{pr(c') \leq pr(c)} pr(c') \leq pr(c)(k - dom(c))$.

Input: query content set C , integer $k < |I|$, the prior probability $pr(\cdot)$ of query content

Output: the reporting probability Pr_C^k

```

1 sort  $C$  according to  $pr(\cdot)$  in descend order;
2 for  $1 \leq i \leq k$  do
3   initiate a list  $L_i$ ;
4  $lcursor = 1$ ;
5  $isDominant = true$ ;
6  $partial, average = 0$ ;
7 initiate an integer array  $Len$  with  $k$  values of 0;
8 for  $1 \leq i \leq |C|$  do
9   if  $isDominant$  then
10    if  $pr(c_i) \geq sum_{pr}/(k+1-lcursor)$  then
11       $L_{lcursor++}.add(new Segment(i, pr(c_i)))$ ;
12    continue;
13  else
14     $isDominant = false$ ;
15     $average = \sum_{i \leq j \leq |C|} pr(c_j)/(k+1-lcursor)$ ;
16  if  $pr(c_i) - partial < average - Len[lcursor]$  then
17     $L_{lcursor}.add(new Segment(i, pr(c_i) - partial))$ ;
18     $Len[lcursor] = Len[lcursor] + pr(c_i)$ ;
19     $partial = 0$ ;
20  else
21     $seg = new Segment(i, average - Len[lcursor])$ ;
22     $L_{lcursor}.add(seg)$ ;
23     $Len[lcursor + +] = average$ ;
24    if  $seg.length == pr(c_i)$  then
25      continue;
26     $seg' = new Segment(i, pr(c_i) - seg.length)$ ;
27     $L_{lcursor}.add(seg')$ ;
28     $Len[lcursor] = seg'.length$ ;
29 while  $L_1 \neq \emptyset$  do
30   initiate a report  $r$ ;
31   add the heads of  $L_1, \dots, L_k$  to  $r$ ;
32    $step = \min_{1 \leq i \leq k} (L_i.head.length/Len[i])$ ;
33   for  $1 \leq i \leq k$  do
34      $j = L_i.head.label$ ;
35      $Pr_C^k(c_j, r) = (step \times Len[i])/pr(c_j)$ ;
36     if  $L_i.head.length == step \times Len[i]$  then
37        $L_i.pop()$ ;
38   else
39      $L_i.head.length -= step \times Len[i]$ ;
40 return  $Pr_C^k$ ;

```

ALGORITHM 1: Multilayer Alignment.

Definition 9 (dominated content). Given the global set C of query contents, an integer $k > 0$, and the prior probability $pr(\cdot)$, $c \in C$ is a dominated content, not a dominant content.

According to the process of segment alignment in MLA, each dominant content takes up an entire layer. If there are remaining layers, the dominated contents take up these layers and none of them take up an entire layer. In the rest of this paper, we use dominant layer and dominated layer to denote a layer taken up by a dominant content or dominated contents, respectively. Recall the instance in Figure 1; c_1 is a dominant content and layer 1 is a dominant layer. Query contents $c_2, \dots, c_7$ are dominated contents and layers 2, 3, and 4 are dominated layers.

Definition 10 (layering strategy). Given the global set C of the query contents, an integer $k > 0$, the prior probability $pr(\cdot)$ of query contents, and a reporting probability Pr_C^k , let the query contents in each report be permuted arbitrarily and denote the i th query content of report r by c_i^r . A layering strategy induced by $\mathcal{M}$ is a k dimensional vector, whose i th component is calculated as $\sum_{r \in R_C^k} Pr_C^k(c_i^r, r) pr(c_i^r)$. When query contents are sorted by the value of $Pr_C^k(c, r) pr(c)$ in descend order, the standard layering strategy is induced.

According to the above definition of layering strategy, a mechanism has multiple layering strategies. Intuitively, a mechanism assigns the prior probability of each query

content to one or multiple reports. A report r contains k parts from distinct query contents, and they can be viewed as k segments on k layers. When query contents in each report are permuted, we can build k layers by connecting all the segments on the same layer from different reports together. To this end, we call the k dimensional vector a layering strategy. Next we define the entropy of a layering strategy.

Definition 11 (entropy of layering strategy). Given a layering strategy $ls = \{v_1, \dots, v_k\}$, the entropy of ls is calculated as $H(ls) = -\sum_{1 \leq i \leq k} v_i \log v_i$.

Lemma 12. Given a mechanism $\mathcal{M}(R_C^k, Pr_C^k)$, let ls be any induced layering strategy of $\mathcal{M}$; then $H_E(\mathcal{M}) \leq H(ls)$.

Proof. Suppose the query contents in each report of M are arbitrarily sorted, and we get an induced layering strategy $ls = \{v_1, \dots, v_k\}$. Denote the set of reports with posterior probability larger than 0 by $R = \{r_1, \dots, r_m\}$, and $c_{i,j}$ is the j th query content of report r_i in the process of inducing ls . For $1 \leq j \leq k$, we have the following equation:

$$\begin{aligned} v_j &= \sum_{1 \leq i \leq m} Pr(c_{i,j}) Pr(r_i | c_{i,j}) \\ &= \sum_{1 \leq i \leq m} Pr(c_{i,j} r_i). \end{aligned} \quad (4)$$

$$\sum_{1 \leq i \leq m} Pr(r_i) = 1.$$

By applying the log-sum inequality [7] (adopted in the last but one line in the below), we have the following condition:

$$\begin{aligned} H_E(M) &= \sum_{1 \leq i \leq m} Pr(r_i) H(r_i) \\ &= - \sum_{1 \leq i \leq m} Pr(r_i) \sum_{1 \leq j \leq k} Pr(c_{i,j} | r_i) \log Pr(c_{i,j} | r_i) \\ &= - \sum_{1 \leq i \leq m} \sum_{1 \leq j \leq k} Pr(c_{i,j} r_i) \log \frac{Pr(c_{i,j} r_i)}{Pr(r_i)} \\ &= - \sum_{1 \leq j \leq k} \sum_{1 \leq i \leq m} Pr(c_{i,j} r_i) \log \frac{Pr(c_{i,j} r_i)}{Pr(r_i)} \\ &\leq \sum_{1 \leq j \leq k} \left(\sum_{1 \leq i \leq m} Pr(c_{i,j} r_i) \right) \left(\log \frac{\sum_{1 \leq i \leq m} Pr(c_{i,j} r_i)}{\sum_{1 \leq i \leq m} Pr(r_i)} \right) \\ &= - \sum_{1 \leq j \leq k} v_j \log v_j = H(ls) \end{aligned} \quad (5)$$

So we prove that $H_E(M) \leq H(ls)$. $\square$

Lemma 13. Given a mechanism $\mathcal{M}(R_C^k, Pr_C^k)$ generated by MLA, ls is the standard layering strategy of $\mathcal{M}$ and $\mathcal{M}'$ is an arbitrary mechanism; then $\mathcal{M}'$ has at least one induced layering strategy ls' satisfying the fact that $H(ls) \geq H(ls')$.

Proof. Given the mechanism $\mathcal{M}$ produced by MLA together with its standard layering strategy ls , we prove Lemma 13 by conducting an induced layering strategy ls' for an arbitrary mechanism $\mathcal{M}'$, so that $H(ls) \geq H(ls')$. To this end, we sort the query contents in each report of $\mathcal{M}'$ as follows.

For each report r of $\mathcal{M}'$, we iterate all the query contents. For a query content c , if it is a dominating query content determined by MLA and its order in ls is $o_{ls}(c)$, we set the order of c in r by $o_{ls}(c)$. After arranging all the dominating query contents, we sort the rest of query contents in r by $pr(c) \times Pr(r | c)$ in descend order and then fill the blanks in the ordering of r . In this way we conduct an inducing layering strategy ls' of $\mathcal{M}'$, and in the following we are to prove that $H(ls) \geq H(ls')$.

Let n_{dom} be the number of dominant layers in ls , and we first investigate the first n_{dom} layers of ls' . For each dominant content on the j th layer of ls , it is also aligned only on the j th layer of ls' . At the same time, on the j th layer of ls' there are possibly dominated contents. So we get that for each dominant layer the length of ls' is no smaller than that of ls , i.e., $ls'.v_j \geq ls.v_j$, $1 \leq j \leq n_{dom}$. As a consequence, the total length of dominated layers in ls' is no larger than that of ls if $n_{dom} < k$; i.e., $\sum_{n_{dom} < j \leq k} ls'.v_j \leq \sum_{n_{dom} < j \leq k} ls.v_j$.

Here we turn to a necessary observation of modifying a layering strategy at two layers with increased entropy. Suppose ls'' is an arbitrary layering strategy, and its values on layer j and layer h are different. With no loss of generality, assume $ls''.v_j > ls''.v_h$. Then we move a length of l from layer j to layer h ; here $0 < l < 2(ls''.v_j - ls''.v_h)$. It is easy to see that the entropy of the modified layering strategy is larger than the entropy of ls'' . Next we transform ls' to ls with a series of modifications of the above type between two layers with different lengths.

The transform includes two phases. In the first phase, ls' make the dominated layers (here the dominated layers and dominant layers are determined by ls) have the same length. Let l_{avg}^{dtd} be the average length of dominated layers for ls' . We repeat the below modification. Each time we pick the dominated layer with smallest length and largest length, and move the length from longer to the shorter until either one of them reaches l_{avg}^{dtd} . Then the number of layers with length l_{avg}^{dtd} increases by at least one. After at most $k - n_{dom}$ modifications, phase 1 terminates. And each modification make the entropy of ls' increase. If the dominated layers of ls' have the same length at the beginning of phase 1, its entropy remains unchanged.

In phase 2, we investigate each of the first n_{dom} layers. For a layer $j \leq n_{dom}$, let the j th dominant content of MLA be c_j^{dom} . Then we remove a length of $ls'.v_j - pr(c_j^{dom})$ and distribute it evenly to dominated layers. After that, the length of each dominated layers for ls' is no larger than that of ls (denoted l_{dtd}). Meanwhile, the remaining length $pr(c_j^{dom})$ of layer j is larger than l_{dtd} . According to the observation above, each modification of phase 2 will increase the entropy of ls' . After at most n_{dom} modifications, ls' will be transformed to ls , and each modification will not decrease the entropy.

Combining phase 1 and phase 2, we conduct a transformation from an induced layering strategy ls' of an arbitrary mechanism to ls , which is an induced layering strategy of the mechanism produced by MLA. Each step of the conducted transformation will increase the entropy or keep the entropy unchanged, so we prove that $H(ls') \leq H(ls)$. $\square$

Lemma 14. *Given a mechanism $\mathcal{M}(R_C^k, Pr_C^k)$ generated by MLA, and ls is the standard layering strategy of $\mathcal{M}$, then $H_E(\mathcal{M}) = H(ls)$.*

Proof. The standard layering strategy of $\mathcal{M}$ restores the result of segment alignment in the process of MLA. Let $L_1, \dots, L_k$ be the lengths of the k generated layers. Due to the shrinking process of MLA, the initiated reports have the same ratios between pairs of corresponding query contents on two given layers. Thus the standard layering strategy ls of $\mathcal{M}$ could be calculated as $ls = \{L_1 / \sum_{1 \leq j \leq k} L_j, L_2 / \sum_{1 \leq j \leq k} L_j, \dots, L_k / \sum_{1 \leq j \leq k} L_j\}$. For each produced report r_i in the process of inducing ls , we use $c_{i,j}$ to denote the j th query content in r_i . Then we have $Pr(c_{i,j})Pr(r | c_{i,j}) / Pr(c_{i,j'})Pr(r | c_{i,j'}) = L_j / L_{j'}$, for $1 \leq j, j' \leq k$. So we can get that the entropy of each report r_i equals the entropy of ls . As a consequence the expected entropy of $\mathcal{M}$ can be calculated as follows:

$$H_E(\mathcal{M}) = \sum_{r \in \mathcal{M}.R} Pr(r) H(r) = \sum_{r \in \mathcal{M}.R} Pr(r) H(ls) = H(ls). \quad (6)$$

So we prove that $H_E(\mathcal{M}) = H(ls)$. $\square$

Lemmas 12, 13, and 14 illustrate the relationship between the expected entropy achieved by MLA and the entropy of induced layering strategies of any other mechanisms. Based on these facts, we get the following theorem.

Theorem 15. *MLA achieves the optimal expected entropy.*

Proof. According to Lemma 14, the mechanism $\mathcal{M}$ produced by MLA achieves the expected entropy of $H(ls)$, where ls is the standard induced layering strategy of $\mathcal{M}$. Assume $\mathcal{M}'$ is an arbitrary mechanism, and it has at least on induced layering strategy ls' so that $H(ls') \leq H(ls)$ due to Lemma 13. At the same time, $H_E(\mathcal{M}') \leq H(ls')$ according to Lemma 12. Then we have $H_E(\mathcal{M}) \geq H_E(\mathcal{M}')$, so we prove that MLA achieves the optimal expected entropy through $\mathcal{M}$. $\square$

Theorem 16. *MLA achieves the optimal dp-coefficient.*

Proof. Given the mechanism $\mathcal{M}$ produced by MLA together with its standard layering strategy ls , we conduct an induced layering strategy ls' for an arbitrary mechanism $\mathcal{M}'$ in the same way as the proof of Lemma 13. We sort the query contents in each report of $\mathcal{M}'$ as follows. For each report r of $\mathcal{M}'$, we traverse its query contents. For a query content c , if it is a dominated content determined by MLA and its order in ls is $o_{ls}(c)$, we set the order of c in r by $o_{ls}(c)$. After

arranging all the dominating query contents, we sort the rest of query contents in r by $pr(c) \times Pr(r | c)$ in descend order, and then fill the blanks in the ordering of r . The first layer of ls only contains the first dominant content; however the first layer of ls not only contains the first dominant content entirely but also possibly dominated contents. So we have $ls.v_1 \leq ls'.v_1$. On the other hand, we know that the total length of dominated layers in ls is no smaller than that of ls' . At the same time, each dominated layer has the same length in ls while the k th layer in ls' has the smallest length. Then we have $ls.v_k \geq ls'.v_k$. In $\mathcal{M}$ the dp-coefficient is actually $\ln(ls.v_1 / ls.v_k)$. Denote the set of reports produced by $\mathcal{M}'$ by $\mathcal{M}'.R_C^k = \{r_1, \dots, r_m\}$, and let $c_{i,j}$ be the j th query content in report r_i , $1 \leq i \leq m$ and $1 \leq j \leq k$. Then we have $\sum_{1 \leq i \leq m} Pr(c_{i,1})Pr(r_i | c_{i,1}) = ls'.v_1$ and $\sum_{1 \leq i \leq m} Pr(c_{i,k})Pr(r_i | c_{i,k}) = ls'.v_k$. Thus $\max_{1 \leq i \leq m} (Pr(c_{i,1} | r) / Pr(c_{i,k} | r)) = Pr(c_{i,1})Pr(r | c_{i,1}) / Pr(c_{i,k})Pr(r | c_{i,k}) \geq ls'.v_1 / ls'.v_k$. The dp-coefficient achieved by $\mathcal{M}'$ is $\ln(Pr(c_{i,1} | r) / Pr(c_{i,k} | r)) \geq \ln(ls'.v_1 / ls'.v_k)$. Since we have got that $ls.v_1 \leq ls'.v_1$ and $ls.v_k \geq ls'.v_k$, we conclude that $\ln(ls.v_1 / ls.v_k) \leq \ln(ls'.v_1 / ls'.v_k) \leq \ln(Pr(c_{i,1} | r) / Pr(c_{i,k} | r))$. That is to say the dp-coefficient of $\mathcal{M}$ is no larger than that of an arbitrary mechanism $\mathcal{M}'$. So we prove that MLA achieves the optimal dp-coefficient through $\mathcal{M}$. $\square$

6. Evaluation

This section evaluates the performance of our proposed MLA algorithm based on three real-life datasets, and evaluation results report the comparison between MLA and three existing approaches including *MEE*, *MER* [8], and *DLS* [3].

6.1. Evaluation Setting

Datasets. To obtain the prior probability of query contents, we employ three real-life datasets including *TX* and *CA* from [9] and *POIs* from [10]. *TX* and *CA* contains street objects in the state of Texas and California. Each object is labeled with a coordinate and a set of keywords. *POIs* contains worldwide coordinates and geotags. We use the coordinates as locations, and take the keywords and geotags as query contents. We divide *TX*, *CA*, and *POIs* into $8km \times 8km$ regions and calculate a prior distribution of query contents for each region. Given the number of query contents $|C|$, we pick query contents with top- $|C|$ frequency and they are used to compute the prior probability. In *TX* and *CA*, some keywords such as city name and state name are removed since they dominate the frequency but provide no meanings. For each dataset, the average measures of its regions are reported in the evaluation results. The details of *TX*, *CA*, and *POIs* are introduced in Table 2.

Testbed. We implement our proposed MLA and competitors including *MEE*, *MER*, and *DLS* in Java language. The JDK version is jdk1.8.0_151. All of the evaluation is conducted on

TABLE 2: Details of datasets in evaluation.

Name	Description	Original/Preprocessed Size	Number of Words
TX	street objects in Texas	14182368/557918	64934
CA	street objects in California	13820481/559349	70584
POIs	POIs worldwide	1157570/1157570	585626

FIGURE 3: Expected entropy versus parameter k .

a PC-machine with i7-7700 CPU, 8GB memory, and 1TB 7200rpm Hard Disk.

Query Generation. For each prior distribution obtained for a region, we generate 1000 queries, which follows the prior distribution, to test *DLS*. The internal loop times is set to 50 as in [3]. For *MLA*, *MEE*, and *MER* we directly evaluate the privacy measures using the mechanism obtained for each prior distribution.

Privacy Measures. We employ three privacy measures to evaluate content privacy achieved by *MLA* and its competitors. These privacy measures are (1) expected entropy; (2) dp-coefficient; and (3) effective k . Expected entropy and dp-coefficient are introduced in Section 3. Effective k measures the number of query contents whose posterior probability is positive, and it measures the uncertainty of the reports in a mechanism.

Parameters. We test the effects of two parameters on the privacy measures we employ. These parameters include the number of query content in a report (denoted k) and the number of query contents in the global set C (denoted N). In the following evaluation k is set to 5, 10, 15, 20, and 25 and its default value is 10. Parameter N is set to 50, 60, 70, 80, 90, and 100 and its default value is 80.

6.2. Evaluation Results. Figures 3 and 4 depict the expected entropy achieved by *MLA* and its competitors. We first study the effects of parameter k on the expected entropy in Figure 3. Here the total number of query contents in

C is set to 80 and k is increased from 5 to 25. As shown in Figure 3, our proposed *MLA* achieves the best expected entropy in the real-life datasets of *TX*, *CA*, and *POIs*. This is consistent with the fact that *MLA* achieves the optimal expected entropy. The achieved expected entropy of *MLA* and its competitors grows with parameter k , since a larger k improves the uncertainty of reports in a mechanism. In the more skewed dataset, i.e., *CA*, *MLA* outperforms *MEE* and *MER* in larger degree than that of the case in datasets of *TX* and *POIs*. The reason is that *MLA* splits larger prior probability of query contents into a larger number of reports; thus it is more suitable to deal with skewed prior distribution of query contents. On the other hand, in *MEE*, *MER* and *DLS* (approximately) keep the ratio of posterior probability for two query contents the same as that of their prior probability. Compared to the datasets of *TX* and *POIs*, the expected entropy achieved in *CA* is smaller correspondingly, since more skewed distribution of query content prior probability decreases the optimal expected entropy.

Figure 4 presents the achieved expected entropy when parameter N grows from 50 to 100 while k is fixed at 10. *MLA* again outperforms its competitors in terms of expected entropy. When N grows, the expected entropy of *MLA*, *MEE*, and *MER* slightly increases while *DLS* gets decreasing expected entropy. The reason is that increased N brings a relief to the skewness of prior distribution of query contents, so *MLA*, *MEE*, and *MER* achieve better expected entropy. However, due to the process of *DLS*, querying top frequent contents will make fewer query contents in reports eliminated. When an increasing N relieves the effects of top frequent contents, more query contents in reports of *DLS* get eliminated. Consistent with what is shown in Figure 3, a larger

FIGURE 4: Expected entropy versus parameter N . FIGURE 5: dp-coefficient versus parameter k .

improvement is obtained in *CA* when we compare *MLA* with *MEE* and *MER*. Meanwhile, better expected entropy is achieved in more uniform datasets of *TX* and *POIs* compared to *CA*.

Next we investigate the dp-coefficient of *MLA* and its competitors. The privacy measure of dp-coefficient depicts the uncertainty of reports in a mechanism. A smaller dp-coefficient means that it is more difficult for the adversary to eliminate a query content from any report. The effects of parameter k on dp-coefficient is studied in Figure 5. We fix parameter N at 80 and increase k from 5 to 25. In all the datasets, *MLA* achieves significantly better dp-coefficient compared to *MEE*, *MER*, and *DLS*. When k increases, the dp-coefficient of all the algorithms grows, since more query contents are packed into the same report. In more skewed dataset, *CA*, a larger dp-coefficient is obtained. The skewness increases the difference of prior probability for query contents in the same report. In datasets *TX* and *POIs*, very small dp-coefficient is achieved when k is set to 5 to 10. For other cases

of k , the dp-coefficient is almost always smaller than 1, and this means very good uncertainty among query contents in any reports. On the other hand, *MEE*, *MER*, and *DLS* suffer a larger dp-coefficient around 4 and 2 in different datasets, respectively.

The effects of parameter N on dp-coefficient are investigated in Figure 6. When we fix k at 10 and increase N from 50 to 100, *MEE*, *MER*, and *DLS* produce nearly constant dp-coefficient. The dp-coefficient of *MEE*, *MER*, and *DLS* in *CA* is larger than 2.5, while a dp-coefficient around 1.5 is obtained for datasets of *TX* and *POIs*. In contrast, the dp-coefficient of *MLA* decreases when N grows, since larger N brings relief to the skewness of prior distribution. In *CA*, *MLA* achieves dp-coefficient smaller than 1.5. For more uniform datasets of *TX* and *POIs*, *MLA* produces very small dp-coefficients. It obtains ideal dp-coefficient with 0 for *TX* dataset, and the dp-coefficient for *POIs* is also very close to 0. This brings significant difficulty to the adversary to infer the actual query content from any report

FIGURE 6: dp-coefficient versus parameter N . FIGURE 7: Effective k versus parameters N and k .

of MLA. Generally speaking, MLA achieves much better dp-coefficient compared to *MEE*, *MER*, and *DLS*, and it is able to produce dp-coefficient close to 0 for more uniform datasets.

Finally we test effective k of MLA and its competitors in Figure 7. Given the value of k and N , the same effective k is obtained for different datasets, so we report the effects of parameter N and parameter k on effective k in Figures 7(a) and 7(b), respectively (not for each dataset individually). As shown in Figure 7, *MLA*, *MEE*, and *MER* achieve the optimal effective k with the value of k . In contrast, *DLS* provides smaller effective k than the value of k . This illustrates the effectiveness of *MLA*, *MEE*, and *MER* with regard to the disability of eliminating any query contents from each report. We argue that an effective k -anonymity mechanism should provide effective k with the value of k .

In summary, *MLA* achieves the best privacy measures of expected entropy, dp-coefficient, and effective k simultaneously, which is consistent with our theoretical analysis in Section 5.

7. Related Work

Privacy issues are attracting more and more attention in people's daily life, and studies for protecting privacy in various fields have been proposed, for instance, [11–13] for social network data, [14] for cloud storage, [15–17] for mobile crowd sensing systems, [18, 19] for wireless sensor networks, [1, 20] for sensory data and devices, [21, 22] for cyberphysical systems, and [23] for IoT applications.

Location privacy and content privacy are recognized in location-based services. Solutions to preserving location

privacy and content privacy in location-based services mainly focus on cloaking technique such as [5]. Cloaking technique employs a third-party server to execute spatial generalization algorithms so that the querier is hidden among at least $k - 1$ users. However, the third-party server unfortunately possibly becomes the single point of failure for privacy or a performance bottleneck of query processing. To this end, a number of client based solutions [2, 3, 6, 8] are proposed recently. Reference [3] works on the problem of generating proper dummies for locations in reported queries to CPS servers for hiding the user's actual locations. In [3], $2k$ locations with similar probability with the user's location are chosen as dummy candidates, and $k - 1$ of them are randomly selected as final dummies. This approach obtains good entropy for the k locations in the reported query. Although this solution includes random nature, the posterior probability of the k reported location is still different due to the process of dummy selection, and the privacy guaranteed is not clear. Reference [6] employs cache to avoid submitting queries to CPS servers as much as possible and thus prevents the leakage of user's location. Reference [2] proposes a mechanism for protecting content privacy in a continuous manner. A set of k query contents are generated for a traveling path, and the user submits the same queries along the path to avoid privacy breach. This fits to continuous querying; however there is no privacy guarantee since it simply chooses query contents with probability larger than a given threshold as candidates. In summary, server-based k -anonymity suffers single point of failure and existing client based solutions do not provide provable privacy guarantee based on the k -location/query contents reported to CPS server. Reference [24] studies improving geoindistinguishability with multiple criteria for better location privacy; however this approach could not be adopted for content privacy due to utility concern. Reference [25] studies protecting privacy for smartphone usage, and this is parallel to our work. Recommendation [26] in location-based system is getting more and more attention, and a location privacy preservation method is proposed for review publication in location-based systems in [27]. The notion of k -anonymity is also developed in statistical databases in [28, 29].

Differential privacy is first introduced in statistic databases [30]. The intuitive idea of differential privacy is that a single change of the input should not modify the output significantly. By this guarantee the adversary cannot recognize the input among all possible inputs similar to the real one. Due to the simple and clean nature of differential privacy, it has been adopted widely, such as machine learning [31], statistic database [32–37], data mining [38], graph [39], data analytic [40], and crowdsourcing [15]. Recent research starts combining correlation [41] and personality [42] nature to original differential privacy. Our work is parallel to the large body of differential privacy research. We combine differential privacy and k -anonymity to provide guaranteed privacy in interactive cyberphysical systems. Differential privacy has been adopted in the literature of privacy protection in location-based services, and [43] ensures that an adversary will not get significant information about a user's location after a query is reported. This is

achieved by making the ratio of two nearby locations' posterior probability similar to that of their prior probability. Mechanisms following or adopting similar privacy guarantee are presented to optimize privacy or utility [44]. Besides k -anonymity and differential privacy, a number of works customize semantic privacy metrics such as [45–47] in social networks. This paper also defines privacy measures based on entropy and differential privacy.

8. Conclusion

This paper investigates preserving content privacy in interactive cyberphysical systems through k -anonymity based mechanisms. We present two privacy metrics denoted expected entropy and dp-coefficient, which are based on entropy and differential privacy, respectively, and formulate the problem of achieving the optimal k -anonymity for content privacy in interactive cyberphysical systems based on these privacy metrics. An algorithm MLA consisting of two phases, namely, segment alignment and mechanism initiation, is proposed to establish mechanisms for achieving the optimal k -anonymity. Theoretical analysis illustrates the attractive property that MLA achieves the optimal expected entropy and the optimal dp-coefficient simultaneously. We conduct evaluation based on three real-life datasets, and three privacy metrics, namely, expected entropy, dp-coefficient, and effective k , which depict uncertainty of reports in mechanisms, are tested. Evaluation result demonstrates that MLA outperforms its competitors including recent client based solutions over all the employed privacy metrics, and these results are consistent with the fact that MLA achieves the optimal k -anonymity for content privacy in interactive cyberphysical systems.

Data Availability

All the data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by Project (Nos. 61602129, 61772157, and U1509216) supported by the National Natural Science Foundation of China; Heilongjiang Postdoctoral Science Foundation Funded Project (Grant No. LBH-Z14118); Sichuan Science and Technology Foundation funded Project (Grant No. 2017JZ0031).

References

- [1] Y Liang I, Zhipeng Cai, Qilong Han, and Yingshu Li, "Location privacy leakage through sensory data," *Security and Communication Networks*, vol. 2017, Article ID 7576307, 12 pages, 2017.

- [2] A. Pingley, N. Zhang, X. Fu, H.-A. Choi, S. Subramaniam, and W. Zhao, "Protection of query privacy for continuous location based services," in *Proceedings of the IEEE INFOCOM 2011*, pp. 1710–1718, Shanghai, China, April 2011.
- [3] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Achieving k-anonymity in privacy-aware location-based services," in *Proceedings of the 33rd IEEE Conference on Computer Communications, IEEE INFOCOM 2014*, pp. 754–762, Toronto, Canada, May 2014.
- [4] L. Sweeney, "k-anonymity: a model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557–570, 2002.
- [5] Y. Wang, D. Xu, and F. Li, "Providing location-aware location privacy protection for mobile location-based services," *Tsinghua Science and Technology*, vol. 21, no. 3, pp. 243–259, 2016.
- [6] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Enhancing privacy through caching in location-based services," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications (IEEE INFOCOM '15)*, pp. 1017–1025, Kowloon, Hong Kong, May 2015.
- [7] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley Series in Telecommunications and Signal Processing, John Wiley & Sons, New York, NY, USA, 2006.
- [8] J. Wang, Y. Li, D. Yang, H. Gao, G. Luo, and J. Li, "Achieving effective k-anonymity for Query privacy in location-based services," *IEEE Access*, vol. 5, pp. 24580–24592, 2017.
- [9] Openstreetmap, <http://cs.sjtu.edu.cn/~yaobin/sas/#dataset>.
- [10] Pois dataset, <http://www.ntu.edu.sg/home/gaocong/datacode.htm>.
- [11] Z. He, Z. Cai, and J. Yu, "Latent-data privacy preserving with customized data utility for social network data," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 665–673, 2018.
- [12] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.
- [13] X. Zheng, G. Luo, and Z. Cai, "A fair mechanism for private data publication in online social networks," *IEEE Transactions on Network Science and Engineering*, 2018, In Press.
- [14] C. Laicheng, L. Yufei, D. Xiaoye, and G. Xian, "User privacy-preserving cloud storage scheme on cp-abe," *Journal of Tsinghua University (Science and Technology)*, vol. 58, no. 2, p. 150, 2018.
- [15] Y. Wang, Z. Cai, G. Yin, Y. Gao, X. Tong, and G. Wu, "An incentive mechanism with privacy protection in mobile crowdsourcing systems," *Computer Networks*, vol. 102, pp. 157–171, June 2016.
- [16] M. Han, J. Li, Z. Cai, and H. Qilong, "Privacy reserved influence maximization in gps-enabled cyber-physical and online social networks," in *Proceedings of the 2016 IEEE International Conferences on Big Data and Cloud Computing (BDCloud)*, pp. 284–292, Atlanta, Ga, USA, 2016.
- [17] M. Han, Y. Li, J. Li, L. Li, and Q. Han, "Maximizing influence in sensed heterogeneous social network with privacy preservation," *International Journal of Sensor Networks*, vol. 1, no. 1, article 1, 2017.
- [18] H. Huang, T. Gong, P. Chen, R. Malekian, and T. Chen, "Secure two-party distance computation protocol based on privacy homomorphism and scalar product in wireless sensor networks," *Tsinghua Science and Technology*, vol. 21, no. 4, pp. 385–396, 2016.
- [19] M. Dapeng, W. Chenye, Y. Wu, W. Wei, X. Shichang, and J. Xiaopeng, "Energy-efficient cluster-based privacy data aggregation for wireless sensor networks," *Journal of Tsinghua University (Science and Technology)*, vol. 57, no. 2, p. 213, 2017.
- [20] Y. Liang, Z. Cai, J. Yu, Q. Han, and Y. Li, "Deep learning based inference of private information using embedded sensors in smart devices," *IEEE Network*, vol. 32, no. 4, pp. 8–14, 2018.
- [21] X. Zheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Follow but no track: privacy preserved profile publishing in cyber-physical social systems," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1868–1878, 2017.
- [22] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, 2018, In Press.
- [23] X. Zheng, Z. Cai, and Y. Li, "Data linkage in smart iot systems: a consideration from privacy perspective," *IEEE Communications Magazine*, 2018, In press.
- [24] S. Oya, C. Troncoso, and F. Pérez-González, "Back to the drawing board: Revisiting the design of optimal location privacy-preserving mechanisms," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1959–1972, Dallas, Tex, USA, October 2017.
- [25] L. Zhang, Z. Cai, and X. Wang, "FakeMask: a novel privacy preserving approach for smartphones," *IEEE Transactions on Network and Service Management*, vol. 13, no. 2, pp. 335–348, 2016.
- [26] Z. Zhou, Z. Cheng, L.-J. Zhang, W. Gaaloul, and K. Ning, "Scientific workflow clustering and recommendation leveraging layer hierarchical analysis," *IEEE Transactions on Services Computing*, vol. 11, no. 1, pp. 169–183, 2018.
- [27] X. Zheng, Z. Cai, J. Li, and H. Gao, "Location-privacy-aware review publication mechanism for local business service systems," in *Proceedings of the IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pp. 1–9, Atlanta, Ga, USA, May 2017.
- [28] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkatasubramanian, "L-diversity: privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data*, vol. 1, no. 1, article 3, Article ID 1217302, 2007.
- [29] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: privacy beyond k-anonymity and l-diversity," in *Proceedings of the 23rd International Conference on Data Engineering*, pp. 106–115, IEEE, Istanbul, Turkey, April 2007.
- [30] C. Dwork, "Differential privacy," in *ICALP 2006: Automata, Languages and Programming*, vol. 4052 of *Lecture Notes in Computer Science*, pp. 1–12, Springer, Berlin, Germany, 2006.
- [31] K. Boyd, E. Lantz, and D. Page, "Differential privacy for classifier evaluation," in *Proceedings of the 8th ACM Workshop on Artificial Intelligence and Security, AISec 2015 - co-located with CCS 2015*, pp. 15–23, Denver, Colo, USA, 2015.
- [32] D. Kifer and A. Machanavajjhala, "No free lunch in data privacy," in *Proceedings of the 2011 ACM SIGMOD and 30th PODS 2011 Conference on Management of Data (SIGMOD)*, pp. 193–204, Athens, Greece, June 2011.
- [33] X. He, A. Machanavajjhala, and B. Ding, "Blowfish privacy: Tuning privacy-utility trade-offs using policies," in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, (SIGMOD '14)*, pp. 1447–1458, Snowbird, Utah, USA, June 2014.
- [34] D. Kifer and A. Machanavajjhala, "Pufferfish: a framework for mathematical privacy definitions," *ACM Transactions on Database Systems (TODS)*, vol. 39, no. 1, article 3, 2014.

- [35] S. Haney, A. Machanavajjhala, and B. Ding, "Design of policy-aware differentially private algorithms," *Journal Proceedings of the VLDB Endowment*, vol. 9, no. 4, pp. 264–275, 2015.
- [36] S. Song, Y. Wang, and K. Chaudhuri, "Pufferfish privacy mechanisms for correlated data," in *Proceedings of the the 2017 ACM International Conference*, pp. 1291–1306, Chicago, Ill, USA, May 2017.
- [37] S. Haney, A. Machanavajjhala, J. M. Abowd, M. Graham, M. Kutzbach, and L. Vilhuber, "Utility cost of formal privacy for releasing national employer-employee statistics," in *Proceedings of the the 2017 ACM International Conference*, pp. 1339–1354, Chicago, Ill, USA, May 2017.
- [38] A. Friedman and A. Schuster, "Data mining with differential privacy," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD-2010*, pp. 493–502, Washington, Wash, USA, July 2010.
- [39] S. P. Kasiviswanathan, K. Nissim, S. Raskhodnikova, and A. Smith, "Analyzing Graphs with Node Differential Privacy," in *TCC 2013: Theory of Cryptography*, vol. 7785 of *Lecture Notes in Computer Science*, pp. 457–476, Springer, Berlin, Germany, 2013.
- [40] X. Wu, F. Li, A. Kumar, K. Chaudhuri, S. Jha, and J. Naughton, "Bolt-on differential privacy for scalable stochastic gradient descent-based analytics," in *Proceedings of the the 2017 ACM International Conference*, pp. 1307–1322, Chicago, Ill, USA, May 2017.
- [41] C. Liu, S. Chakraborty, and P. Mittal, "Dependence makes you vulnerable: differential privacy under dependent tuples," in *Proceedings of the Network and Distributed System Security Symposium*, San Diego, Calif, USA, 2016.
- [42] Z. Jorgensen, T. Yu, and G. Cormode, "Conservative or liberal? Personalized differential privacy," in *Proceedings of the 2015 31st IEEE International Conference on Data Engineering, ICDE 2015*, pp. 1023–1034, Seoul, South Korea, April 2015.
- [43] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: differential privacy for location-based systems," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '13)*, pp. 901–914, Berlin, Germany, November 2013.
- [44] N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Optimal geo-indistinguishable mechanisms for location privacy," in *Proceedings of the 21st ACM Conference on Computer and Communications Security, CCS 2014*, pp. 251–262, Scottsdale, Ariz, USA, November 2014.
- [45] Z. He, Z. Cai, J. Yu, X. Wang, Y. Sun, and Y. Li, "Cost-efficient strategies for restraining rumor spreading in mobile social networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 2789–2800, 2017.
- [46] Z. He, Z. Cai, and X. Wang, "Modeling propagation dynamics and developing optimized countermeasures for rumor spreading in online social networks," in *Proceedings of the 35th IEEE International Conference on Distributed Computing Systems (ICDCS '15)*, pp. 205–214, Columbus, Ohio, USA, July 2015.
- [47] Z. He, Z. Cai, Y. Sun, Y. Li, and X. Cheng, "Customized privacy preserving for inherent data and latent data," *Personal and Ubiquitous Computing*, vol. 21, no. 1, pp. 43–54, 2017.