An Efficient and Robust Privacy Protection Technique for Massive Streaming Choice-based Information

Ji Zhang*
Faculty of Health, Engineering and Sciences&
Centre for Systems Biology
University of Southern Queensland, Australia
Ji.Zhang@usq.edu.au

Xuemei Liu
Faculty of Business and Law
University of Southern Queensland, Australia
Xuemei.Liu@usq.edu.au

Yonglong Luo†
School of Mathematics and Computer Science
Anhui Normal University, China
ylluo@ustc.edu.cn

ABSTRACT

Protecting users' privacy when transmitting a large amount of data over the Internet is becoming increasingly important nowadays. In this paper, we focus on the streaming choice-based information and propose a novel anonymization technique for providing a strong privacy protection to safeguard against privacy disclosure and information tampering. Our technique utilizes an innovative two-phase encoding-and-decoding approach which is very easy to implement, highly efficient in terms of speed and communication, and is robust against possible tampering from adversaries. The experimental evaluation demonstrates the promising performance of our technique.

Keywords

Privacy protection, anonymization, choice-based information

1. INTRODUCTION

We had witnessed an astronomical growth of the amount of data being transferred over the Internet these days. It is critical to ensure that the privacy of users are properly protected when data are transferred, shared and utilized. Data anonymization has proven to be an effective means to significantly reduce the risk of privacy disclosure and information tampering.

Much research work has been conducted on information anonymization. Based on the scope of data being anonymized, we can broadly classify anonymization methods into two major categories, *i.e.*, the table-based anonymization and the tuple-based anonymization. The table-based anonymization methods, represented by k-anonymization [3] [4] [5], performs anonymization on all the tuples in the table, so that each tuple is identical with at least $k-1$ other tuples in the table. Many variants of k-anonymization, such as t-closeness [1] and l-diversity [2], are also proposed to ensure that each tuple is highly similar, rather than identical, with at least $k-1$ other tuples in the table. In contrast, the tuple-based anonymization methods perform anonymization on every sensitive attribute of each tuple in order to generate k distinct values for every sensitive attribute within the tuple [6] citeMISA11 [8] [10]. By increasing the value cardinality of the sensitive information by k times, it becomes much more difficult for adversaries to figure out the true value of the sensitive information. In view of this, the table-based k-anonymization is quite different from the tuple-based k-anonymization despite both of them are termed k-anonymization.

One particular type of data we will focus in this work are those carrying choice-based information. Choice-based information is very common in our daily life which involve picking up a choice from a predetermined finite set of candidates. In this paper, we only consider the simplest case that each data instance is only associated with one choice from the candidates. Much of the choice-based information is generated through various real-time applications where the sensitive information needs to anonymized in a distributed manner and then transmitted in the form of data streams to a central processor for decoding and further processing. Examples of these applications include electronic voting, online questionnaire/surveying, Participatory Sensing Systems (PSS) and information sharing on social media, to name a few.

The table-based anonymization methods are effective to safeguard userąŕs privacy, but they show some major limitations in dealing with these real-time applications. First, there is inevitable information loss and distortion in their anonymization process that are not acceptable by many of these applications. Second, their anonymization process is irreversible in the sense that it is not trivial to obtain the original data by simply utilizing the anonymized result. Finally, the data streams generated by these applications make it extremely difficult to achieve the objective of k - anonymization (*i.e.*, each data instance is identical to other $k-1$ instances). The tuple-based anonymization, in comparison, has many unique advantages that can well solve the limitations of the table-based anonymization. The tuple-based anonymization can be decoded to retrieve the original data

*Dr Zhang is the correspondence author.

†Dr Luo is the correspondence author.

CIKM'13, Oct. 27–Nov. 1, 2013, San Francisco, CA, USA.

http://dx.doi.org/10.1145/2505515.2507816.

without any information loss or distortion. Also, the objective of k-anonymization can be easily achieved for streaming data because each data is anonymized individually.

There have been some proposed methods for tuple-based anonymization for real-time systems that are suitable for dealing with massive streaming choice-based information [6] [7] [8] [10]. They are based on a subset encoding technique to provide privacy protection when sensitive information is transmitted from distributed agents to the central processor. However, they still suffer from a vulnerability called targeted decoding which is referred to as the success in decoding of the true encoding position for a given candidate. Like the full decoding, the targeted decoding may still be very harmful. For example, there are likely some leading and popular voting candidates in an election. If adversaries are able to figure out the true sequence position for the leading candidates then they can tamper the votes in order to make it more favorable to one leading candidate or more unfavorable to another one. In response to this, we propose a new technique in this paper to solve this problem. Specifically, the technical contributions of this paper are summarized as follows:

1. We propose an innovative tuple-based anonymization method for dealing with a large amount of streaming choice-based information. Our technique provides effective anonymization for the choice-based information and can solve the problem of targeted decoding suffered by the existing method;
2. The processes of anonymization and de-anonymization are very efficient which enables our technique to be scalable for large-scale applications;
3. Our technique is more capable and flexible in catering for the desired level of security requirement by adapting the length of the so-called Base Encoding Sequence. This can contribute to achieving the best possible trade-off between privacy protection and communication overhead encountered;
4. The preliminary experimental evaluation conducted shows that our technique is efficient in terms of both speed and communication overhead and more robust than the existing method against targeted decoding and tampering.

For ease of presentation and readers' understanding, we use the case of *voting in an election* to present our technique in this paper.

2. CONCEPTS AND DEFINITIONS

Definition 2.1 *Base Encoding Sequence*: The Base Encoding Sequence is a sequence of consecutive integer numbers in the range of [1, N_L], where N_L is the length of the Base Encoding Sequence. Mathematically, the Base Encoding Sequence can be presented as $baseSequence = \{i | i \text{ is an integer AND } 1 \leq i \leq N_L\}$.

Base Encoding Sequence serves as the basis for encoding the valid and dummy votes in our work, where valid votes are the authentic votes casted by voters for selecting a particular candidate while the dummy votes are those generated artificially by the system for privacy preserving purpose. The Base Encoding Sequence contains both the true and false encoding positions of candidates.

Definition 2.2 *True encoding positions of candidates:* A position within the Base Encoding Sequence will be chosen for each candidate as its true encoding position, which is used to represent the candidate in the transmission process. Every candidate will be assigned one and only one distinctive true encoding position. Thus, the number of true encoding positions is equal to the number of candidates N_C.

Definition 2.3 *False encoding positions of candidates*: All the other positions that are not chosen as the true encoding positions of candidates are called their false encoding positions. The number of possible false encoding positions of candidates is equal to $N_L - N_C$.

3. AN OVERVIEW OF OUR TECHNIQUE

3.1 Stage 1: Generate and Decode the Candidate Encoding Sequences

In the first stage of our technique, the candidate encoding sequences are generated and transmitted to the central decoder to obtain the true encoding positions of all the candidates.

First, let us introduce the candidate encoding sequences. Once the true encoding positions of all the candidates in the Base Encoding Sequence have been selected, we can then apply the following idea of anonymization to encode the true encoding positions of candidates into candidate encoding sequences. The encoding sequence of a candidate C_i contains its name, one true encoding position and a number of false encoding positions, which can be presented in the following format

$$S(C_i) = \{C_i, truePosition(C_i), falsePosition_j(C_i)\}$$

where $truePosition(C_i)$ is the true encoding position of C_i and $falsePosition_j(C_i)$ is the j^{th} false encoding position of C_i, $1 \leq j \leq m$. In practice, the elements in the candidate encoding sequences, except the candidate names, will be arranged randomly when the sequences are transmitted to the central decoder. The generation of candidates encoding sequences follows the following three rules: 1) The true encoding position of a given candidate is in the encoding sequence of this candidate only, and will not appear in encoding sequences of other candidates; 2) Any false encoding positions will appear at least twice amongst all the candidates encoding sequences; 3) There are no any duplicate true encoding positions or false encoding positions in the encoding sequence of any given candidate.

As an example, let's suppose $N_L = 10$ and the encoding sequences of three candidates, A, B and C are $S(A) = \{A, 1, 5, 3, 9\}$, $S(B) = \{B, 2, 3, 8, 6\}$, and $S(C) = \{C, 1, 7, 6, 8, 9\}$. The true encoding positions for A, B and C are $\{5\}$, $\{2\}$, $\{7\}$, respectively because these positions only appear once amongst all the three encoding sequences, while $\{1, 3, 6, 8, 9\}$ are the false encoding positions as they appear multiple times among all the encoding sequences.

The decoding algorithm performed at the central decoder is even simpler than the encoding algorithm executed in the distributed encoders. A simple counting process is performed to extract the true and false encoding positions of candidates from the candidate encoding sequences. Those positions in the Base Encoding Sequence which is only occupied by a single candidate is extracted as its true encoding position. Formally, a position $p \in baseSequence$ is decoded as the true encoding position of a candidate $C_i \in C$ if

$count(p) = 1$ and $p \in S(C_i)$. Other positions are decoded as the false encoding positions.

Our technique shares the same desirable feature as methods proposed in [6] [7][8] that there is no any sensitive information concerning how the candidates are encoded being directly sent from the encoders to the central decoder. Such sensitive information is collectively decoded in the central decoder using all the candidate encoding sequences received through different channels. This strategy is a very advantageous because even adversaries manage to compromise some transmission channels, it is still very difficult for them to decode the true encoding positions of all the candidates.

3.2 Stage 2: Encode and Decode Votes

3.2.1 Encoded and Decode Valid Votes

After the candidate encoding sequences have been decoded at the central decoder, it is ready to decode the votes receiving from different channels. The term "valid votes", referred to the votes cast by voters, is used here in order to distinguish from the term "dummy votes" (which is to be discussed in the next subsection). The most important information each valid vote will contain is the true encoding position of the candidate selected by the vote. Other less important information included in the vote may include the identification and other biographical information of the voter and the date/time and district location of the vote, etc.

Upon receiving a valid vote, the central decoder will evaluate the position number contained in the vote. If the position number matches the true encoding position of a particular candidate(which has been available in the first stage), this vote is counted as a vote cast to that candidate. Otherwise, the vote is deemed illegitimate and will be discarded.

3.2.2 Use of Dummy Votes

From the above discussion on vote encoding, we know that the possible distinctive position numbers appearing in valid votes is equal to the number of candidates. Given the typically small number of candidates, using valid votes only in the transmission process will lead to a small search space for adversaries to figure out the true encoding position for each candidate. In order to solve this problem, the so-called dummy votes are created purposely to confuse adversaries and heightens the difficulties for decoding. Each dummy vote carries a false encoding position number together with other information consistent with the valid votes, with some information being fictional if necessary.

The dummy votes can be easily identified by the central decoder because of the false position number they carry. The identification process involves a lookup of the true encoding positions of all candidates. If the position number on the vote does not match the true encoding positions of any candidates, then the vote is labeled as dummy and discarded as a result.

Let N_{valid} denote the number of valid votes for all candidates. The expected number of dummy votes to be generated is $(\frac{N_L}{N_C} - 1) * N_{valid}$. The basic idea is to ensure that the ratio between the valid and dummy votes is consistent with the ratio between the true and false encoding positions in the Base Encoding Sequence. In the micro scale, approximately $\frac{N_L}{N_C} - 1$ dummy votes are generated following each valid vote. However, the exact number of dummy votes associated with each valid vote is governed by a normal distribution $\mathcal{N}(\mu, \sigma)$ with $\mu = \frac{N_L}{N_C} - 1$ and $\sigma = \sigma_{valid} * \frac{N_L - N_C}{N_C}$.

3.2.3 Detection of Tampered Votes

Our method is a very robust in detecting tampered votes. Tampered votes are those whose position number is purposely changed by adversaries. The key information that adversaries want to tamper is the position number of the candidate appearing in the encoded votes. Vote tampering can be categorized into the following four possible cases: **Case 1**: a valid vote is changed to a dummy vote; **Case 2**: a dummy vote is changed to a valid vote; **Case 3**: a valid vote cast to a candidate is changed to a valid vote for another candidate; **Case 4**: a dummy vote is changed to another dummy vote.

Amongst the above four possible cases, our technique can handle Case 1 and 4 effectively. For Case 2 and 3 nevertheless, our system will fail to detect the tampered votes as they are totally indistinguishable with the true valid votes. Despite this, by utilizing a Based Encoding Sequence with an appropriate length, we are able to guarantee the chance of our system failing to detect tampered votes in Case 2 and 3 is maintained in an acceptably low level. The theory is presented in Proposition 3.1 and Corollary 3.2 (The proof is omitted due to space limit but is available upon request).

Proposition 3.1 The possibility that our technique fails to detect tampered votes is $\frac{N_C}{N_L}$.

Corollary 3.2 The minimum length of the Base Encoding Sequence, guaranteeing that the chance of failing to detect a tampered vote is no higher than pro, is $\frac{N_C}{pro}$.

4. EXPERIMENTAL RESULTS

In this section, we will carry out experimental evaluation to investigate the performance of the technique we proposed in this paper. The experimental evaluation will not only evaluate the efficiency and robustness of our proposed technique but also conduct comparison between our method and the encoding scheme proposed in [6][7][8], which is referred to as the competitive method in the experiment.

4.1 Efficiency Study

We first evaluate the efficiency of our technique, with a focus on how efficient the vote encoding and decoding can be performed. The execution time of our technique are presented in Figure 1 and 2, respectively. They show that our technique performs very efficiently as all of the execution time grows in a linear order with respect to the number of votes handled. In addition, the speed for encoding and decoding votes of our technique is comparable with those of the competitive method.

4.2 Robustness Study

For the first stage, we evaluate the number of candidates whose true positions can be accurately decoded when some transmission channels are compromised by adversaries. As Figure 3 demonstrates, our method is very robust as adversaries need to compromise all the channels in order to decode the true encoding positions for all candidates and 80% of the channels to eventuate decoding for 50% of the candidates. In comparison, the competitive method is much vulnerable than our method as the decoding for all the candidates can be easily done by simply analyzing a relatively

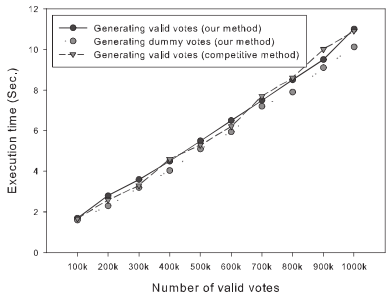

Figure 1: Speed of encoding votes

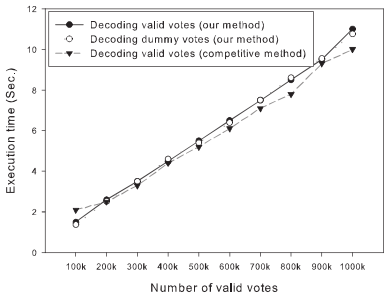

Figure 2: Speed of decoding votes

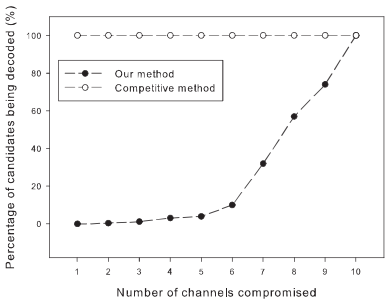

Figure 3: Percentage of the candidates being decoded

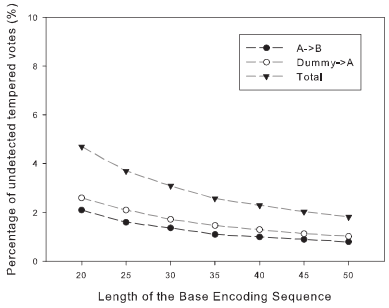

Figure 4: Percentage of tampered votes fail to be detected

small percentage (i.e., 10%) of votes through even a single channel.

For the second stage, we concentrate on the likelihood that our technique fails to detect tampered votes. The experiments are designed to randomly change the encoded position of a vote (either a valid or dummy vote) to another random number within the scope of the Base Encoding Sequence. The tampered votes is set to occupy 10% of the total number of votes. The result presented in Figure 4 shows that such tampering is barely effective - less than 3% of votes on average can be tempered without being detected, showing that tampering does not pose any significant threats to the encoded information. A higher N_L value will further reduce the risk of the system to an even lower level.

5. FUTURE RESEARCH DIRECTIONS

In the future, we will explore several possible improvements on our technique. First, we will investigate the possibility of reducing the number of dummy votes without compromising the level of privacy protection, which can contribute to achieving a better communication performance. Second, new algorithms are to be developed to better distinguish these two kinds of votes, whereby we can have a better idea regarding the time and location when information tampering occurs, and corresponding actions and investigation can be carried out in a more timely and targeted manner.

6. ACKNOWLEDGMENT

The authors would like to acknowledge the support received for this research work from Australian Digital Futures Institute, University of Southern Queensland, through its Research Leadership Development Program and the Digital Futures CRN (Collaborative Research Network) program.

7. REFERENCES

[1] N. Li, T. Li and S. Venkatasubramanian. t-Closeness: Privacy Beyond k-anonymity and l-diversity. *ICDE 2007*: 106-115.

[2] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. l-Diversity: Privacy beyond k-anonymity. *ICDE 2006*.

[3] P. Samarati and L. Sweeney. Generalizing data to provide anonymity when disclosing Information. *PODS 1998*.

[4] P. Samarati. Protecting respondentsạ́r identities in microdata release. *IEEE TKDE*, 13(6), 2001.

[5] L. Sweeney. k-Anonymity: A Model for Protecting Privacy. *International Journal on Uncertainty Fuzziness Knowledge-based Systems*, 10(5), 2002

[6] M. Murshed, T. Sabrina, A. Iqbal, and K. Alam, A Novel Anonymization Technique to Trade-off Location Privacy and Data Integrity in Participatory Sensing Systems. *NSS'10*, 2010.

[7] M. Murshed, A. Iqbal, T. Sabrina, and K. M. Alam, A Subset Coding based k-Anonymization Technique to Trade-off Location Privacy and Data Integrity in Participatory Sensing Systems. *NCA'11*, May 2011.

[8] A. Iqbal, T. Sabrina, M. Murshed, M. Ali. Verifiable and Privacy Preserving Electronic Voting with Untrusted Machines. *IEEE TrustCom-13*, 2013.

[9] K. Huang, S. Kanhare, and W. Lu, Preserving privacy in participatory sensing systems. *Computer Communications*, 2009.

[10] L. Hu, Shahabi C, Privacy Assurance in Mobile Sensing Networks: Go Beyond Trusted Servers, *IEEE Intl. Conf. on Pervasive Computing and Communications*, 2010.