PHDP: Preserving Persistent Homology in Differentially Private Graph Publications

Tianchong Gao[1], Feng Li[1]
[1]Indiana University-Purdue University Indianapolis, Indianapolis, IN, U.S.A.
tgao@iupui.edu, fengli@iupui.edu

Abstract

Online social networks (OSNs) routinely share and analyze user data. This requires protection of sensitive user information. Researchers have proposed several techniques to anonymize the data of OSNs. Some differential-privacy techniques claim to preserve graph utility under certain graph metrics, as well as guarantee strict privacy. However, each graph utility metric reveals the whole graph in specific aspects.

We employ persistent homology to give a comprehensive description of the graph utility in OSNs. This paper proposes a novel anonymization scheme, called PHDP, which preserves persistent homology and satisfies differential privacy. To strengthen privacy protection, we add exponential noise to the adjacency matrix of the network and find the number of adding/deleting edges. To maintain persistent homology, we collect edges along persistent structures and avoid perturbation on these edges. Our regeneration algorithms balance persistent homology with differential privacy, publishing an anonymized graph with a guarantee of both. Evaluation result show that the PHDP- anonymized graph achieves high graph utility, both in graph metrics and application metrics.

Index terms: Online social network; privacy and utility; differential privacy; persistent homology.

This is the author's manuscript of the article published in final edited form as:

Gao, T., & Li, F. (2019). PHDP: Preserving Persistent Homology in Differentially Private Graph Publications. IEEE INFOCOM 2019 - IEEE Conference on Computer Communications, 2242–2250. https://doi.org/10.1109/INFOCOM.2019.8737584

I. INTRODUCTION

Online social networks (OSNs) have a lot of incentives to share user data with third parties. This sharing can enable targeted advertisements, friendship or connection recommendations, and the analysis of collaboration between researchers. When OSN data is shared, various anonymization techniques can be employed to preserve the privacy of OSN users.

Differential-privacy mechanisms are widely used because they provide a strong privacy guarantee without the assumption of background knowledge of the attackers [7]. These mechanisms are applied to different abstraction models to preserve different kinds of graph information. For instance, the dK-2 series model [20] stores the information as pairs of node degrees. The published graph maintains the degree distribution of the original graph. The Hierarchical Random Graph (HRG) model stores a cluster of nodes in the same branch of the HRG tree [22]. The HRG model has superior performance in preserving clustering information.

Although these mechanisms claim to preserve graph utility under some specified utility metrics, the true utility of the published graphs is questionable for two reasons: First, the chosen metrics are limited by the graph abstraction models. Previous studies have shown that none of the mechanisms have good performance under all the metrics [9]. Second, existing metrics only describe the graph in a certain angle. For example, while the degree distribution and the clustering coefficient disjointedly reveal the graph utility in two specific aspects, each aspect does not cover the other. Thus, lots of useful graph information gets lost or distorted during the graph anonymization process, especially when the anonymization mechanisms are based on these types of graph metrics.

In this paper, persistent homology is employed to analyze graph utility. Persistent homology tracks the topological features of the whole graph at different distance resolutions in different dimensions [11]. Unlike the well-studied utility metrics, persistent homology gives a comprehensive summarization of the graph. Since persistent homology is a novel utility metric, the main challenge of our anonymization scheme is to extract the corresponding persistent homology information and preserve it in the published graph.

First, our scheme model the OSN by an adjacency matrix for two reasons: (1), the adjacency matrix contains the same topological information as the distance matrix. Because the persistent homology filtering phase tracks the persistent structures with different distances, the structures in distance matrix can be easily map to the ones in adjacency matrix. (2), the adjacency matrix has less sensitivity in edge adding or deleting than other graph abstraction models, i.e., it requires less noise under the same privacy level.

Second, to preserve the persistent homology in OSNs, we analyze the theoretical meaning behind the barcodes. We find that the graph data are squished when calculating the barcodes, which is different from existing studies of point cloud data [2, 19]. Squishing complicates the analysis of high-dimensional holes but also opens the opportunity to extract the actual shapes of the persistent structures in OSNs. While original persistent homology defines 3-D voids in H_2 bars, 4-D voids in H_3 bars, etc., these high dimensional voids are squished into special kinds of 2-D holes. Therefore, preserving the polygons (holes in OSNs) defined by the barcodes is preserving persistent homology.

Third, we design an anonymization algorithm which preserves the holes and satisfies differential privacy. The holes occupy a small part of the network; differential privacy is maintained through modifying the other parts. Particularly, we divide the adjacency matrix into four kinds of sub-matrices, according to the corresponding subgraphs with or without holes. Then different regeneration algorithms are employed to each kind of matrix for the purpose of satisfying differential privacy and preserving the holes at the same time.

Because holes are the persistent structures, preserving the holes opens a new angle to balance differential privacy with persistent homology. Existing mechanisms aim to preserve the high degree nodes, the communities, and the clusters in the network, but pay no attention to the holes. However, a hole, i.e., nonexistent relationships in an area, also contains meaningful information describing the network. Our PHDP scheme emphasizes the holes. We consider other structures as the links between holes, and these structures are determined when holes are fixed.

The major technical contributions of this study are the following: (1) introducing a novel utility metric, persistent homology, in the analysis of OSNs, (2) proposing the PHDP scheme to balance differential privacy and persistent homology in graph anonymization, and (3) evaluating the PHDP scheme with two real-world datasets and comparing it with other anonymization schemes.

II. Preliminaries

In this paper, an online social network graph is modeled as an unweighted undirected graph $G = (V, E)$, where V is the set of vertices and E is the set of edges.

A. DIFFERENTIAL PRIVACY

Differential privacy is designed to minimize the chance of record identification. When modeling OSNs, the records are the edges. Differential privacy requires that an adversary cannot detect if an edge exists in the original network with high confidence. Two OSNs with at most one edge difference are called neighbor graphs. Sensitivity is defined as follows:

Definition 1

(SENSITIVITY). *The sensitivity* (Δf) *of a function* f *is the maximum distance of any two neighbor graphs in the* ℓ_1 *norm.*

$$\Delta f = \overset{max}{G_1, G_2} \parallel f(G_1) - f(G_2) \parallel \qquad (1)$$

In our scheme, the function f outputs the number of edges added or deleted $(f0$ and $f1)$. If only considering the upper triangle of the adjacency matrix, adding or deleting one edge causes an increase or decrease of 1 in $f0$ or $f1$, which means $\Delta f = 1$.

Definition 2

(ϵ −DIFFERENTIAL PRIVACY). *A randomized algorithm* A *achieves* ϵ *−differential privacy if for all neighbor graphs,* G_1 *and* G_2*, and all output values,* $S \subseteq Range(A)$.

$$Pr[A(G_1) \in S] \leq e^{\epsilon} \times Pr[A(G_2) \in S] \qquad (2)$$

Equation (2) calculates the probability that two neighbor databases have the same result, which means the adversary cannot differentiate them under the same algorithm. Based on this definition, researchers developed the exponential mechanism to achieve ϵ − differential privacy when the result is an output space. It smoothes the original distribution by exponentiating the probabilities with respect to the sensitivity, Δf, and the desired privacy parameter, ϵ. [22]

Theorem 1

(EXPONENTIAL MECHANISM). *For a function* $f: (G, OS) \rightarrow$*, the randomized algorithm* A *that samples an output* O *from* OS *with the probability proportional to* exp $(\frac{\epsilon \cdot f(G,OS)}{2\Delta f})$ *achieves* ϵ *−differential privacy.*

where OS is the output space containing all the outputs.

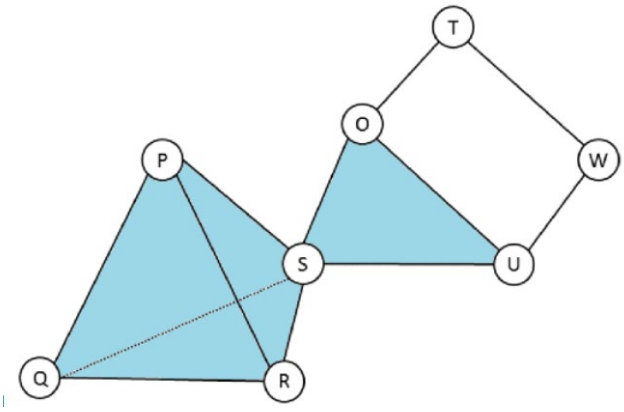


Fig. 1: An example of the simplicial complex

B. Persistent homology

Persistent homology is a utility metric that summarizes the graph in multi-scales. Simply speaking, persistent homology is a summarization of holes along different dimensions and different δ, where δ is the distance to build margins of holes. In real world cases, we call an object as a hole when it has bounding margin(s) but it has no plane region, e.g., the circles and the polygons. When defining persistent homology, we follow this rule but extend it to other dimensions.

Persistent homology is presented in the form of barcodes, which have two parts. The Vietoris-Rips (VR) simplicial complex describes the structural change at different spatial resolutions in one dimension, while the Betti number describes the dimensions [3].

VR simplicial complex. Persistent homology is based on the simplicial complex. A simplicial complex set K contains points, line segments, triangles and high-dimension components. K satisfies the following conditions:

1. Any face of a simplex from K is in K, where the face of K_n is the convex hull of the non-empty subset of the $n+1$ points, which define K_n.w
2. The intersection of any two simplices, $\sigma 1, \sigma 2 \in K$, is either $\emptyset$ or a face of both $\sigma 1$ and $\sigma 2$.

In a simplicial $k - complex$, the highest dimension of simplices is k. For instance, the $1 - complex$ is the line segment, the $2 - complex$ is the convex hull of the triangle and the $3 - complex$ is the convex hull of the tetrahedron.

The VR complex is one of the abstract models of the simplicial complex. It introduces the distance parameter δ, and then forms the simplicial complex set K, such that for all node pairs $(v_i, v_j) \in K$, the distance between v_i and v_j is less than or equal to δ.

Fig. 1 shows an example with one 3-complex, one 2-complex and some 1-complexes. The node set {O,U, T,W} has no 2-complex because the pairwise distance within O-W and T-U both are above δ.

Barcode. While the VR complex is defined on the specific δ, persistent homology chooses various δ and gives an increasing sequence of VR complexes.

$$K_0 \subseteq K_1 \subseteq \cdots \subseteq K_n = K \qquad (3)$$

Persistent homology collects the features in a wide range of distance and gives a comprehensive description of the structure.

Through applying Betti numbers, the persistent homology overcomes the restriction of dimension. The Betti number Betti_n gives the number of (n+1)-dimensional holes. Particularly, Betti_0 is the number of connected components, Betti_1 is the number of holes and Betti_2 is the number of voids.

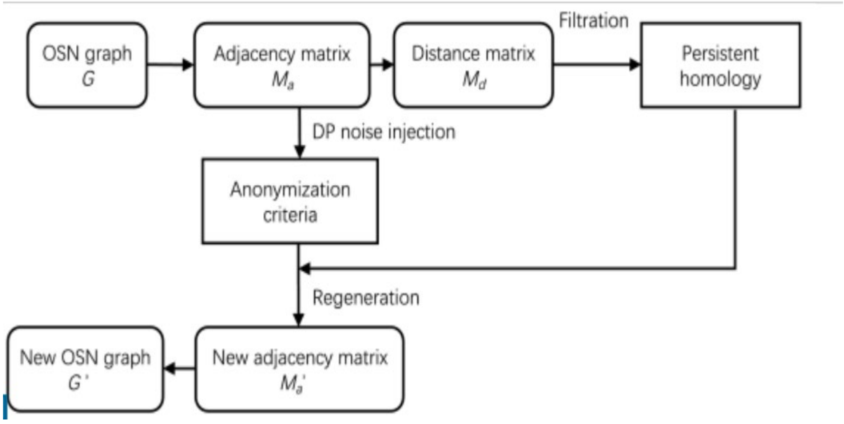


Fig. 2: Scheme overview

In n dimension, the vector space of n-holes is represented by H_n, which can be calculated by the n-cycles and n –boundaries [24]. This calculation gives the Betti intervals to describe the homology of H_n. These intervals are called barcode, where each interval means a component or a hole in the corresponding dimension [11]. For example, the H_1 interval [1, 2) in Fig. 3 is related with the H_1 hole {P, Q, S, R}. The intervals begin with the δ the holes born. And they end with the δ the holes die. The intervals show the birth time and death time of the components. In conclusion, the barcode collects the information of the existing periods of all components and holes when changing the distance δ.

In OSNs, a 2-D hole is a polygon with at least 4 sides. Because there is no ideal circle in the OSN, polygons are appropriate for the common definition of holes, which has a circular boundary and the inside is empty. The high-dimensional holes are the voids, which have triangles as surface and an empty interior. A polygon with at least 4 sides implies that all nodes on the polygon have at least one node which is not directly connected, while the triangles have all nodes pair-wisely connected. When we increase the δ, there are more edges in the network, which may both forming and filling, i.e., adding and deleting, holes.

III. SCHEME

Given an OSN G, our goal is to publish an anonymized network G' that maximally preserves persistent homology while satisfying ϵ −differential privacy. The general idea of the PHDP scheme is to preserve the persistent structures in the anonymized graph. Fig. 2 shows the structure of the scheme. Section III-A describes how the OSN graph is modeled as an adjacency matrix and the corresponding distance matrix. Section III-B describes the division of the adjacency matrix into four types depending on if the corresponding subgraph has holes. Afterwards, the PHDP scheme applies an MCMC procedure to output the number of flips required to achieve differential privacy. Section III-C describes the application of different regeneration algorithms to the varying types of sub-matrices in order to preserve existing holes and prevent the creation of new ones.

Anonymizing user identity without perturbing the graph structural information leaves the OSN vulnerable to potential de-anonymization attacks [16]. Hence, the PHDP scheme includes both the naive ID removal and the differential-privacy topological anonymization. Since the published graph G' contains no identity information, the original vertex label is trivial. Only the graph topology information is applied with anonymization and utility preservation.

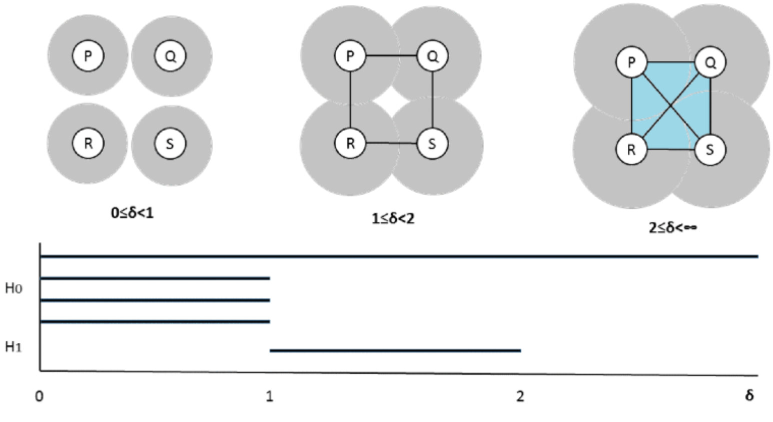


Fig. 3: An example of the barcode

A. SYSTEM MODEL

The PHDP scheme employs the adjacency matrix model. Compared to the other graph abstraction models, e.g., dK-2 and HRG, the adjacency matrix model has the least sensitivity $(\Delta f = 1)$. Because the differential privacy noise is proportional to sensitivity, the resulting adjacency model has the least distortion. Another reason for choosing the adjacency matrix model is that it can be easily transformed to a distance matrix—the input for barcode extraction.

To link the adjacency matrix with the distance matrix, we must first establish the definition of distance. Given an unweighted OSN graph G, the most direct definition of distance δ is the length of the shortest path between a pair of users. Following that definition, persistent homology is captured based on the distance matrix. Fig. 3 gives an example of the barcode. In the original network, four nodes P, Q, S, R form a square. When $\delta < 1$, there are no edges in the graph. Each node is a component in H0, so there are four bars in [0, 1). When $\delta \geq 1$, the nodes are connected together to form a component and this component exists until the end. So there is one bar of H_0 in [1,∞). When $\delta < 2$, the node pairs P-S and R-Q are not connected. Then the four nodes form a hole in 2-dimension. So there is one bar of H_1 in [1, 2).

Obviously, under this definition of distance, there is an equivalent relationship among the graph topology, the adjacency matrix M_a and the distance matrix M_d. For example, we can use Ma to build an isomorphic graph of the graph G. We can also traversing the edges to generate the distance matrix.

The process of capturing persistent homology is a filtration in M_d. Taking Fig. 3 as the example, we have

$$M_a = \begin{array}{|c|c|c|c|}\hline 0 & 1 & 0 & 1 \\ \hline 1 & 0 & 1 & 0 \\ \hline 0 & 1 & 0 & 1 \\ \hline 1 & 0 & 0 & 0 \\ \hline \end{array} \qquad M_d = \begin{array}{|c|c|c|c|}\hline 0 & 1 & 2 & 1 \\ \hline 1 & 0 & 1 & 2 \\ \hline 2 & 1 & 0 & 1 \\ \hline 1 & 2 & 1 & 0 \\ \hline \end{array}$$

Equation (3) suggests that filtration is also employed among three adjacency matrices with $\delta = 0{,}1{,}2$. And $\delta = 0$ is omitted because M_a is a zero matrix and $K_0 = \emptyset$.

$$\left\{ K_1 \in \begin{array}{|c|c|c|c|}\hline 0 & 1 & 0 & 1 \\ \hline 1 & 0 & 1 & 0 \\ \hline 0 & 1 & 0 & 1 \\ \hline 1 & 0 & 1 & 0 \\ \hline \end{array} \right\} \subseteq \left\{ K_2 \in \begin{array}{|c|c|c|c|}\hline 0 & 1 & 1 & 1 \\ \hline 1 & 0 & 1 & 1 \\ \hline 1 & 1 & 0 & 1 \\ \hline 1 & 1 & 1 & 0 \\ \hline \end{array} \right\} = K$$

The example shows that given δ, the new graph connects all the node pairs with distance less than or equal to δ in the original graph.

B. ANONYMIZATION

With the adjacency matrix M_a, Algorithm 1 has two phases of anonymization. The first phase is dividing: Ma splits into sub-matrices according to the barcode. The second phase is noise injection: the number of flips of 0s and 1s in the sub-matrix are calculated based on the differential-privacy criteria. Then in the regeneration sub-scheme, the position of the 1s preserves the persistent homology.

Dividing. In the dividing phase, the nodes are divided into different groups according to the barcodes they involved. The input of the dividing algorithm is the whole graph and the corresponding adjacency matrix. The output is a node sequence, placing nodes in the same group adjacent to each other.

In order to preserve the H_0 bars, we need to locate the connected components and extract the corresponding nodes. When $\delta = 0$, each node in the graph is a component. The number of H_0 bars equals the number of nodes, which is trivial because it equals the size of the adjacency matrix. When $\delta \geq 1$, the number of H_0 bars equals to the number of disconnected subgraphs. For preserving these H_0 bars, the adjacency matrix should have a node label sequence that groups the nodes according to the subgraphs they belong to.

In order to protect the H_1 and H_2 bars, we need to locate the holes in the network. The nodes involved in each bar are extracted based on the Morse Theory [18]. In particular, for each 2-D and 3-D hole, the 2-D boundaries and 3-D boundaries are extracted. The nodes belonging to the same boundary, i.e., the same hole, are grouped together. If the original graph has disconnected subgraphs, each hole is contained in a single subgraph. Hence, grouping them together does not violate the previous grouping result. Fig. 1, for example, has one subgraph (itself) and one hole. The new node sequence can be {{O, T, U, W}, {P, S, Q, R}}.

The two steps of grouping give a new vertex label sequence and the corresponding adjacency matrix Ma. Then Ma is divided into four kinds of sub-matrices according to the node groups of barcodes.

- $M0$, which only contains 0, shows there are 0 edges between two disconnected subgraphs.

- M1, whose nodes are extracted from H_1 and H_2 bars, shows the edges within a hole component (in Fig. 1, the nodes {O, T, U, W}).

Algorithm 1 Anonymization algorithm

```
Input: Adjacency matrix $M_a$, privacy budget ϵ.Output: Number of flips f0 and f1 in each sub-matrix
Get the barcodes, group nodes correspond with each bar.
Get a sequence of node labels, rebuild $M_a$ with the sequence.
Divide $M_a$ into sub-matrices according to groups.
for each sub-matrix in $M_a$ do
    Apply the MCMC procedure, get the distribution of (f0,f1).
    Apply the exponential mechanism, sample (f0,f1)=(a,b) with probability exp(ϵ·p(a,b)2).
end for
return f0, f1 for each sub-matrix.
```

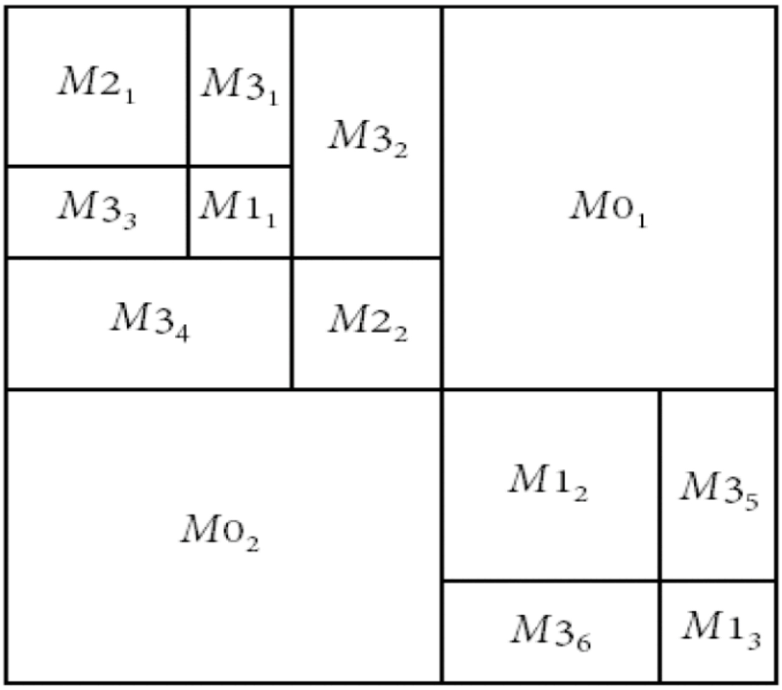


Fig. 4: An example of dividing the M_a

- *M2*, whose nodes are not extracted from barcode, shows the edges in a subgraph without holes (in Fig. 1, the nodes {P,S, Q, R}

- *M3* shows the edges between the nodes from M1 and M2 matrices (in Fig. 1, the nodes {O, U, S}

From the description, M1 and M2 are matrices on the diagonal; M0 and M3 are not on the diagonal. Fig. 4 shows the M_a corresponding to a simple graph with two disconnected subgraphs. One subgraph has one H_1 bar (in $M1_1$) while the other has two H_1 bars (in $M1_2$ and $M1_3$).

Noise injection. In this phase, each sub-matrix is perturbed to satisfy the differential-privacy criteria. The perturbed matrix M'_a is graphic, meaning it can regenerate a graph, if and only if M'_a has the following three properties: First, M'_a only contains Os and Is. Second, M'_a is symmetric. Third, the values on the diagonal of M'_a are a110, because self-loop is not allowed in OSNs.

Hence, only one of two symmetric matrices needs the anonymization. For instance, in Fig. 4, the algorithm perturbs $M3_1$ then copies it to $M3_3$. When the sub-matrix is on the diagonal, only the upper triangle of the matrix is perturbed. The row number and column number of a matrix is represented by h and w. Then to a sub-matrix not on the diagonal, the effective size $S = h \cdot w$. To a sub-matrix on the diagonal, the effective size S is the size of the upper triangle (except the diagonal), which is $\frac{h2-h}{2}$.

Because the basic step of graph anonymization is edge addition or deletion when the differential privacy is defined on edges, we use two numbers $f0$ and fl to model the anonymization process. $f0$ shows the number of Os flipping to Is, and fl shows the number of Is flipping to 0s. It requires the data structure, i.e., the adjacency matrix, to store the 0s and Is.

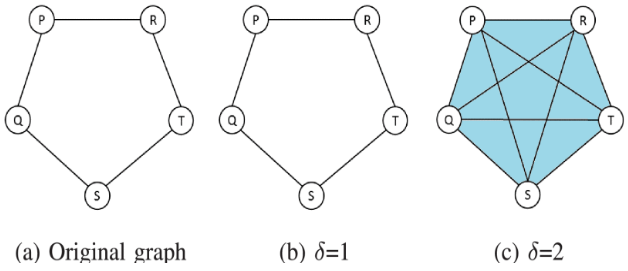
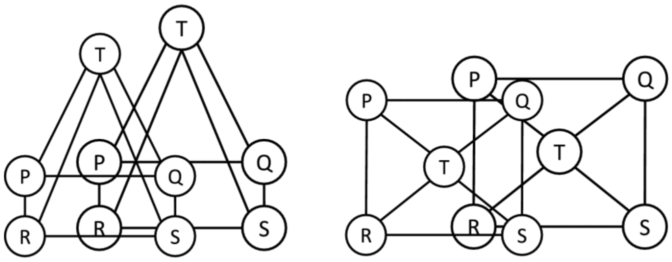


Fig. 5: Example of 2-D hole in H1



(a) Squishing result 1 (b) Squishing result 2

Fig. 6: Example of none 2-D hole in H_1

In order to achieve $\epsilon-$differential privacy, we apply the exponential mechanism to the adjacency matrix of the graph. However, the exponential mechanism requires the natural distribution of $(f0, f1)$, then it calculates the two numbers $f0$ and $f1$. Thus, we employ a Markov Chain Monte Carlo (MCMC) procedure to obtain the approximate natural distribution. MCMC is a class of algorithms for sampling from a probability distribution [12]. After the Markov chain reaches its stationary distribution, the subsequently visited states of the chain can be used to simulate the natural distribution.

In our work, the states of the Markov chain are adjacency matrices and the neighbor states are two adjacency matrices with one number difference, i.e., one edge adding or deleting. Particularly, beginning from the original sub-matrix, each step of the Markov chain has the following sub-steps: (1) Uniformly and randomly choose one out of S numbers in the adjacency matrix. (2) Flip that number, i.e., 0 changes to 1 or 1 changes to 0, and build the new adjacency matrix. (3) Compare the new matrix with the very beginning matrix to get the numbers of flips $f0$ and $f1$; record the two numbers. (4) Go to the next MCMC step.

Applying the MCMC procedure described above from zero to a large step size, we can find the approximate distribution of $(f0, f1)$. Particularly, the probability of a target pair $(f0, f1) =$

(a, b) is denoted as $p(a,\ b)$, which is the number of times f0=a and f1=b divided by the total number of steps of the Markov chain.

Finally, the exponential mechanism is embedded with the MCMC procedure to satisfy differential privacy. Specifically, the Markov chain is the same. Instead of the unperturbed probability $p(a,\ b)$, the algorithm samples $(f0, f1) = (a, b)$ with perturbed probability exp$\left(\frac{\epsilon \cdot p(a,b)}{2\Delta f}\right)$, where $\Delta f = 1$

C. REGENERATION

The regeneration sub-scheme designs algorithms to choose the 1s and 0s to flip, which preserves the persistent homology as well as satisfies the requests of $f0$ and $f1$. Although all sub-matrices have the corresponding flipping numbers, the four kinds of sub-matrix $M0, M1, M2, M3$ have different regeneration algorithms. For the matrices representing the barcodes, i.e., $M0$ and $M1$, we need to preserve the structures in it. For the matrices not directly representing the barcodes, i.e., $M2$ and $M3$, we have more freedom to change edges.

M0. In order to preserve H_0, the $M0$ matrix has a strict restriction that all values in it are 0. Although the $M0$ matrix does not produce any fl, the regenerated matrix cannot consume any $f0$ or fl either. Similarly, the $f0$ and fl consumption in $M1$ matrices is also restricted because the holes need to be preserved. Hence, $M2$ and $M3$ matrices consume all the $f0$ and fl in $M0$.

M1. The $M1$ matrices are related with $H1$ and $H2$ bars. According to the definition of persistent homology, the barcode in H_n shows the $(n+1)$-dimensional holes. Fig. 3 and Fig. 5 give examples of 2-dimensional holes, which both have a [1, 2) bar in H_1. Their distance matrices are $M_d(4)$ is a square. $M_d(5)$ is a pentagon. The distance matrices suggest that the necessary condition of a $2-D$ hole (H_1) existing is that δ is less than the maximum value in M_d.

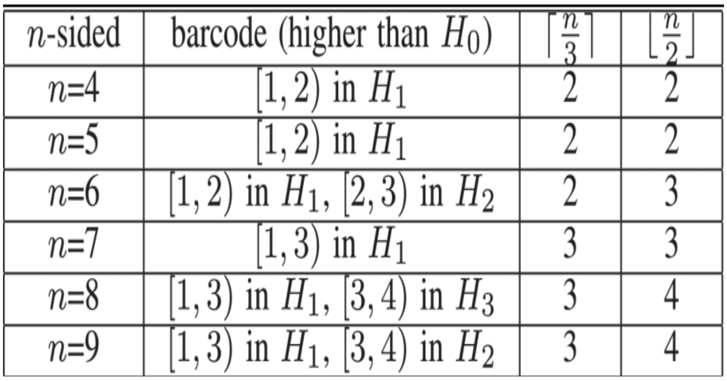
$$M_d(4) = \begin{array}{|c|c|c|c|}\hline 0 & 1 & 2 & 1 \\ \hline 1 & 0 & 1 & 2 \\ \hline 2 & 1 & 0 & 1 \\ \hline 1 & 2 & 1 & 0 \\ \hline \end{array} \qquad M_d(5) = \begin{array}{|c|c|c|c|c|}\hline 0 & 1 & 2 & 2 & 1 \\ \hline 1 & 0 & 1 & 2 & 2 \\ \hline 2 & 1 & 0 & 1 & 2 \\ \hline 2 & 2 & 1 & 0 & 1 \\ \hline 1 & 2 & 2 & 1 & 0 \\ \hline \end{array}$$

When $\delta = 0$, no edges are formed. Therefore, the birth time of H_1 bars is no less than 1. The 2-D hole is defined as a polygon with at least 4 sides. In OSNs, since the holes are in the form of polygons, we use polygons as the basic hole structure to analysis barcodes.

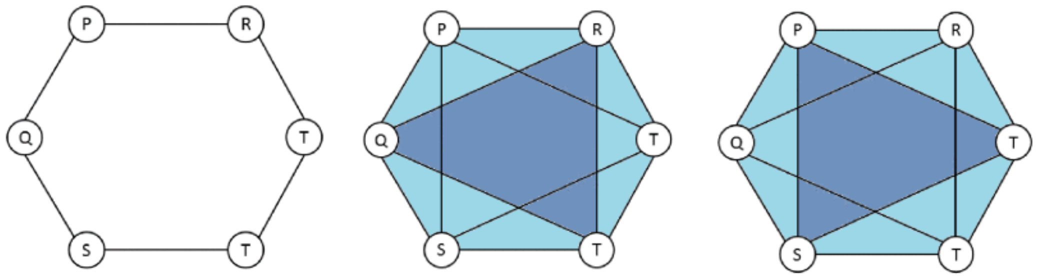
Unlike other data structures with fixed positions for each node, OSNs only define the relationships between nodes. Consequently, OSNs have the possibility called squishing. In a 2-D view of the OSN, a node can be put inside or outside a hole with different squishing results, which influences the existence of the hole. Taking Fig. 6 as the example, in the first result, the square {*P,Q,S,R*} is a 2-D hole. However, in the second result, there are no holes because all the components are triangles. Considering all the squishing results, there are no H_1 bars in the barcode of Fig. 6. The necessary and sufficient condition of H_1 bar existing is that under a specific δ, in all squishing results, there is at least one area which is not filled with a triangle.

The persistent homology also has the ability to capture high-dimensional holes. For example, a sphere with the surface and a void is a simple 3-D hole. When it squishes to 2-D, there should be two layers of surface overlapping with each other. A 3-D hole is inferred from the two layers of surface and there is an H_2 bar. The hexagon in Fig. 7 is another example. The maximum distance is 3 in M_d. When $\delta = 2$, there are two squishing results with the surface filled by triangles. Thus the H_1 bar dies and the H_2 bar exists when $\delta = 2$. And its barcode is in Table I.

TABLE I Barcodes of Polygons



n-sided	barcode (higher than H_0)	$\lceil \frac{n}{3} \rceil$	$\lfloor \frac{n}{2} \rfloor$
n=4	$[1,2)$ in H_1	2	2
n=5	$[1,2)$ in H_1	2	2
n=6	$[1,2)$ in H_1, $[2,3)$ in H_2	2	3
n=7	$[1,3)$ in H_1	3	3
n=8	$[1,3)$ in H_1, $[3,4)$ in H_3	3	4
n=9	$[1,3)$ in H_1, $[3,4)$ in H_2	3	4



(a) $\delta = 1$, original graph (b) $\delta = 2$, squishing 1 (c) $\delta = 2$, squishing 2

Fig. 7: Example of 3-D hole in H_2

We do experiment on the barcodes of polygons, as shown in Table I. Because polygons having more than 7 sides rarely exist in OSNs, the H_0, H_1 and H_2 bars are suitable to rep-resent the persistent structures. Having the mapping between persistent structures and the barcodes, we can generate $M1$ matrices without changing the barcodes.

Similar to $M0$, the regeneration of the $M1$ matrices also has the restriction that no edge is added or deleted. However, edge exchanging makes it possible to reduce $f0$ and fl at the same time inside the $M1$ matrices. Simply speaking, exchanging one of two edges' end nodes, e.g., replacing the edge **P-Q**, **R-T** with the edge **P-T**, **Q-R**, results in reducing $f0$ and $f1$ by two. In our scheme, we model a n-sided polygon with a length-n sequence. Different permutation of the n nodes represents different linking relationships, but they all represent a n-sided polygon.

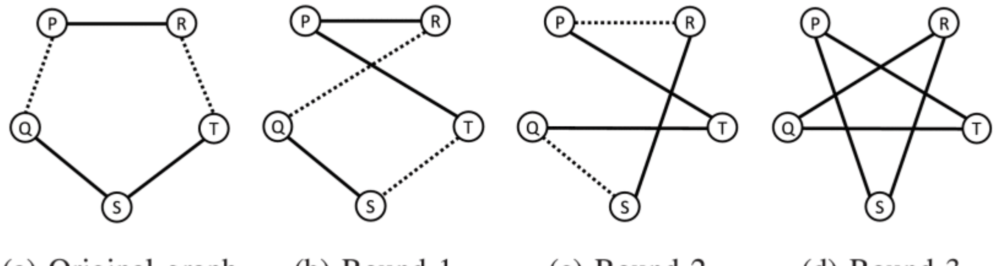
After examining all the permutations, we choose the one who lets $f0$ or $f1$ reduce to zero or reduce $f0$ and fl as much as possible.

Observing Table I, we also make a hypothesis about the high dimensional holes. For a polygon with n sides $(n > 3)$, it has an H_1 bar $[1,\ \lceil\frac{n}{3}\rceil)$. When $\lceil\frac{n}{3}\rceil < \lfloor\frac{n}{2}\rfloor$, it has at least one bar $[\lceil\frac{n}{3}\rceil,\ \lfloor\frac{n}{2}\rfloor)$ in high dimension (higher than H_1). Several properties related to the hypothesis are analyzed in Section IV.

M3. The purpose of regenerating the $M3$ matrices is to avoid creating new holes while adding or deleting edges. As shown in Fig. 9(a), the $M3$ matrices capture the edges between two structures, denoted by A and B. The nodes in A that connect to B form the set A^* The nodes in B that connect to the node A_i forms the set A_i^*. In the basic example, $\{A2, A3, A4\}, A_2^* = \{B2, B3\}$.

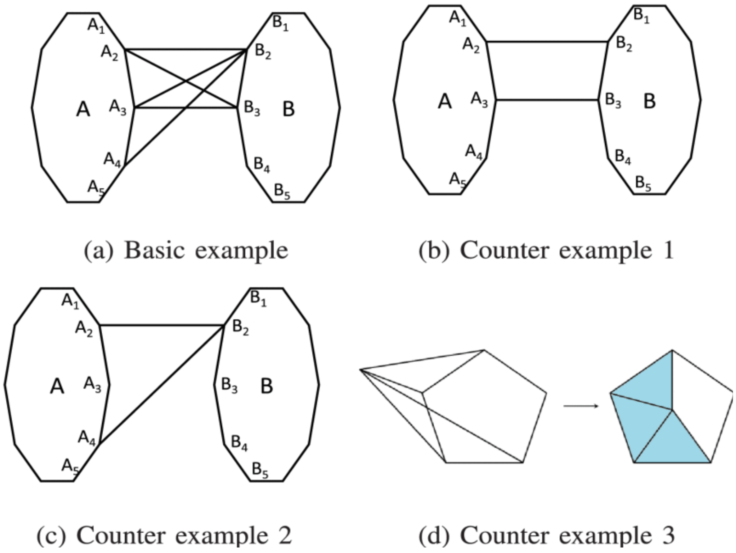
When the regeneration step successfully preserves the bar-codes, the M_3 matrices and corresponding edges should obey the following rules:

1. The nodes in $A*$ should be adjacent to each other, i.e., $\forall A_i \in A^*, \exists Aj \in A^*, (A_i, A_j) \in E$.
2. Sorting the sequence of A_i^* according to size in non-decreasing order $A_i^*, A_j^*, \dots, A_k^*$, the sequence should have $A_i^* \subseteq A_j^* \subseteq \cdots \subseteq A_k^*$.



(a) Original graph (b) Round 1 (c) Round 2 (d) Round 3

Fig. 8: Example of edge exchanging steps



(a) Basic example

(b) Counter example 1

(c) Counter example 2

(d) Counter example 3

Fig. 9: Examples of edges in M3 matrices

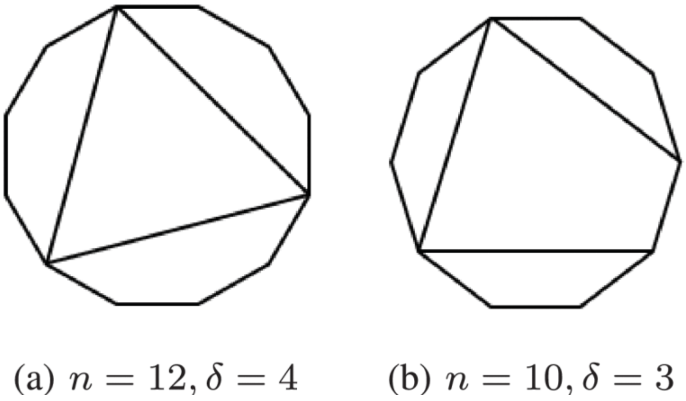
3. The structure belonging to M_1 should have at most three nodes to connect to the other structure. For instance, if A is a hole, then $|A *| \leqslant 3$, where $|A *|$ shows the cardinality of the set $A *$

The examples of violating these three rules are shown in Fig. 9(b), 9(c), and 9(d), respectively. When the nodes in $B *$ are not adjacent, they simply create polygons with more than three sides. When A_i^* and A_j^* both have exclusive nodes, they also create polygons. When $|A *| = 4$, according to the second rule, there is at least one node in B connecting the four nodes in A. Then a polygon with n sides becomes a polygon with $n - 1$ sides, and the barcode has been changed.

The edges are added or deleted based on the three rules. In particular, when deleting edges, our scheme chooses the smallest set A_i^* and deletes the nodes in the set, ensuring $A_i^* \subseteq A_j^*$. After this step, our scheme chooses A_j^* and resumes the same deleting process. When adding edges, our scheme begins from adding nodes to the largest set A_k^*. Furthermore, A^* is restricted to three nodes if A belongs to $M1$.

When structure A contains both hole components and non-hole components, our scheme should not choose more than three nodes that originally belonged to the $M1$ matrices. Taking the $M3_2$ in Fig. 4 as the example, we can first add three nodes from $M1_1$ to A^*. And if we want more edges, we can also add nodes from $M2_1$ to A^*, but nodes in A^* should be connected.

M2. Given the rules of the $M3$ matrix, regenerating $M2$ matrices becomes simple. Intuitively, an $M2$ matrix can be divided into two $M2$ matrices and two $M3$ matrices. And the two $M2$ matrices can be further divided until the size of each matrix is only one. Because the diagonal value of $M2$ matrices should all be 0, regenerating the $M2$ matrix can be viewed as regenerating a group of $M3$ matrices. In these $M3$ matrices, both A and B contain no holes. So only the first and second properties need to be considered in the regeneration.



(a) $n = 12, \delta = 4$ (b) $n = 10, \delta = 3$

Fig. 10: Comparing δ with $\frac{n}{3}$

After regenerating all the sub-matrices, we combine them together to form M′a, and use M′a to build the new graph G′.

IV. ANALYSIS

A. Privacy

Property 1.

The anonymization algorithm achieves ϵ edge differential privacy.

Proof.

In the MCMC procedure, the true distribution of $(f0, f1)$ is well approximated when the total step number is large enough. Then applying the exponential mechanism in sampling gives the $(f0, f1)$ under differential privacy.

The conquer-and-divide procedure achieves differential pri-vacy when $f0$ and fl of the super-matrix is calculated and satisfied. Because differential privacy is only concerned about the value $f0$ and $f1$, dividing the $(f0, f1)$ into different matrices does not violate the differential privacy criteria for the super-matrix.

Furthermore, the persistent structure has the characteristic of preventing identity leakage. The polygons are the basic structure of persistent homology bars. The nodes on a polygon are isomorphic to each other. Also the barcode often has a group of the same bars. Hence, preserving the barcodes does not mean revealing the identity.

B. High-dimensional holes

Two properties related to the hypothesis of high-dimensional holes follow:

Property 2.

For a polygon with n *nodes, holes do not exist when* $\delta \geqslant \lfloor\frac{n}{2}\rfloor$.

Proof.

In a polygon with n nodes, the maximum pairwise distance is $\lfloor\frac{n}{2}\rfloor$. When $\delta \geqslant \lfloor\frac{n}{2}\rfloor$, all pair of nodes are connected, and all holes die.

Property 3.

For a polygon with n *nodes, there are no* H_1 *holes existing when* $\delta \geqslant \lceil\frac{n}{3}\rceil$.

Proof.

According to the definition of H_1 holes, it exists if and only if there is at least one area not filled with triangles in 2-D. So there is at least one polygon that has more than three sides. Fig. 10(a) shows the case of $\delta = \frac{n}{3}$. After linking the edges of length $\frac{n}{3}$, the remaining part, i.e., the polygon in the middle, is a triangle. Fig. 10(b) shows the case of $\delta > \frac{n}{3}$. The remaining part is a tetragon, and its diagonals have length greater than $\frac{n}{3}$. Because δ is restricted to integers in this paper, the upper limit of H_1 hole is $\lceil\frac{n}{3}\rceil$.

Intuitively, a high-dimensional hole exists only if the low-dimensional surface is complete. Considering a ball, when there is a hole on its surface, the void inside is broken. Then we have the hypothesis that high dimensional-holes only exist when $\delta \in [\lceil\frac{n}{3}\rceil, \lfloor\frac{n}{2}\rfloor)$.

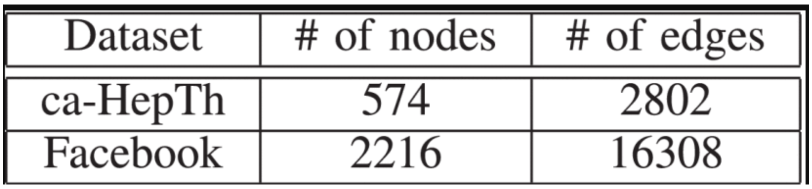
V. EVALUATION

The proposed scheme aims to preserve the persistent structures in the OSN. However, the ultimate impact of the persistent homology on the utility of the graph needs further validation through evaluation. The evaluation is based on the Facebook dataset, and the ca-HepPh dataset [17]. The detailed information is shown in Table II. The barcode extraction program is based on Perseus [18].

The dK-2 graph model, which differential privately pre-serve the dK-2 series in OSN anonymization, is employed to compare with PHDP [20]. Moreover, we also apply the Erdó's-Rényi (E-R) graph model to anonymize the graph, in which the edge present probability equals to the average edge present probability in the original graph [8]. The two differential privacy schemes, dK-2 and PHDP, are compared under the same differential privacy level $\epsilon = 10$. Furthermore, PHDP is also evaluated under a strict privacy level of $\epsilon = 1$. According to Equation

(2), here the same ϵ means the same probability that the adversary can differentiate two neighboring databases, i.e., the same anonymization power. Although PHDP and dK-2 have different graph abstraction models, i.e., adjacency matrix and joint degree matrix, the definition of differential privacy gives us the opportunity to uniformly compare the privacy.

TABLE II Network Dataset Statistics



Dataset	# of nodes	# of edges
ca-HepTh	574	2802
Facebook	2216	16308

A. BARCODES

The first part of the evaluation is to validate the ability to preserve persistent homology of the schemes. Fig. 11 report the persistent barcodes of the ca-HepPh dataset and we obtain the same result in the Facebook dataset. Although all four anonymized graphs have more H_1 or H_2 bars, PHDP has much less distortion in barcodes. In the original ca-HepPh graph, there are 16 H_1 bars and 1 H_2 bar. PHDP $(\epsilon = 10)$ performs the best in preserving the bars: there are 22 H_1 bars and 1 H_2 bar. The PHDP $(\epsilon = 1)$ result has 28 H_1 bars and 3 H_2 bars. The dK-2 result has 300 H_1 bars and 17 H_2 bars. The H_2 bars are [3, 4) which implies that the anonymized graph has a 9-sided polygon. The E-R result has 591 H_1 bars and 49 H_2 bars.

The Facebook barcodes show a similar distribution. In the original graph, there are 185 H_1 bars and 28 H_2 bars. And the two numbers are 314 and 71 in the PHDP $(\epsilon = 10)$ result, 327 and 58 in the PHSP $(\epsilon = 1)$ result, 1142 and 76 in the dK-2 result, and 1688 and 21 in the E-R result. The increase of the H_1 and H_2 bars suggests that there are more holes in the anonymized graph. The users ‘on hole’ are farther apart than the users ‘on non-hole’. While PHDP is confirmed to preserve the persistent homology information under differential privacy criteria, the utility of dK-2 and E-R anonymized graphs is questionable because of the injected holes.

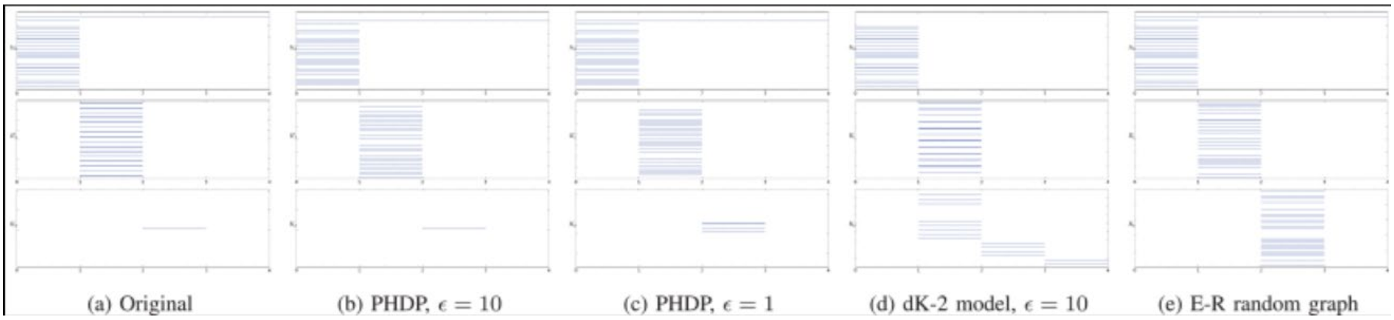
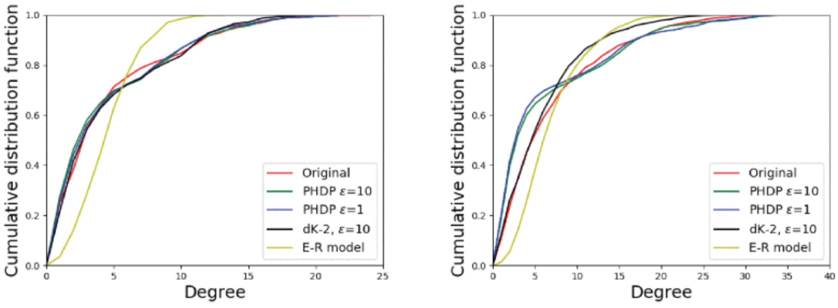

(a) Original (b) PHDP, $\epsilon = 10$ (c) PHDP, $\epsilon = 1$ (d) dK-2 model, $\epsilon = 10$ (e) E-R random graph

Fig. 11: Barcodes of the ca-HepPh graph



(a) ca-HepPh (b) Facebook

Fig. 12: Degree distribution

B. UTILITY METRICS

To demonstrate the links between preserving persistent structures and graph utility, the performance of published graph under utility metrics are compared. The evaluation includes two graph utility metrics, the degree distribution and the clustering coefficient, and one application utility metric, the influence maximization.

Degree distribution. Degree distribution is the number of connections of nodes among the graph. Fig. 12 shows the degree distribution of the two datasets. The PHDP and dK-2 anonymized graphs match the degree distribution of the original graph. Compared to the original ca-HepPh graph, the degree distribution of the PHDP result (ϵ=10) has a root-mean-square error (RMSE) of 0.018, the PHDP result (ϵ=1) has a RMSE of 0.022, the dK-2 result (ϵ=10) has a RMSE of 0.018, but the E-R result has a RMSE of 0.110. Compared to the original Facebook graph, the PHDP (ϵ=10), PHDP (ϵ=1), dK-2 and E-R results have RMSE of 0.053, 0.062, 0.036 and 0.072.

The dK-2 anonymized graph maintains a similar degree distribution because it stores the paired degree information. The PHDP anonymized results suggest that the persistent homology information may have a soft impact on the degree. Intuitively, the holes restrict the edges. The E-R model only has the information of the average degree.

Clustering coefficient. The clustering coefficient shows the level of nodes clustering together. Fig. 13 is the clustering coefficient of the two datasets. Only the PHDP anonymized graphs preserve some clustering information. The original ca-HepPh graph has an average clustering coefficient of 0.52, and it is 0.40 to PHDP (ϵ=10), 0.37 to PHDP (ϵ=1), 0.16 to dK-2 (ϵ=10) and 0.09 to E-R. The average clustering coefficients of the original Facebook graph, the PHDP (ϵ=10) result, the PHDP (ϵ=1), the dK-2 result and the E-R result are 0.45, 0.33, 0.30, 0.13 and 0.10, respectively.

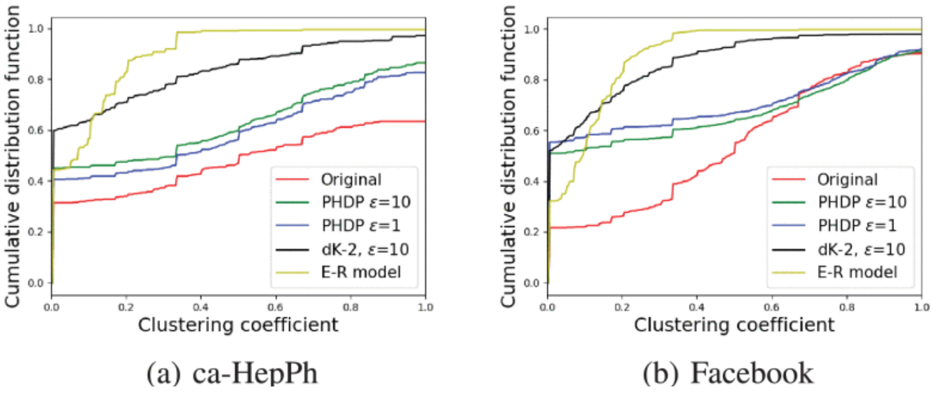
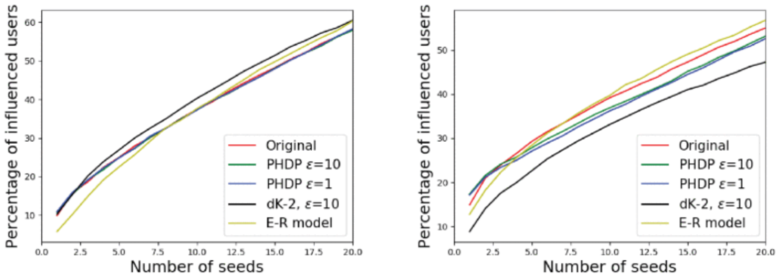


Fig. 13: Clustering coefficient distribution



(a) ca-HepPh

(b) Facebook

Fig. 14: Percentage of influenced users

As shown in the evaluation of the barcodes, in a increasing order of number of holes the graphs are the original graph, the PHDP (ϵ=10), the PHDP (ϵ=10), the dK-2 and the E-R results. This order is also the decreasing order of clustering coefficient. It implies that holes occupy the position of clusters and then decrease the clustering coefficient. While holes are the opposite of clusters, the PHDP anonymized graphs preserve the clustering information by protecting the holes. When all holes in the graph are established, then remaining parts can be filled with clusters. In a real OSN, this shows that the number of holes are less than the number of clusters. Hence, storing holes opens a novel angle to maintain the graph structure.

Influence maximization. Influence maximization [15] is an application that first chooses the important users as seeds, then uses the seeds to influence other people. In the evaluation, a greedy algorithm based on the independent cascade model [6] is employed to choose the seeds who have the most ability to broadcast information. Then the percentage of influenced users are compared among different anonymized graphs, with the same propagation probability, 0.2.

Fig. 14 shows the percentage of influenced users. Although all four anonymized graphs achieve a similar data with the original graph, the PHDP results outperform the other. Com-pared to the original ca-HepPh data, the PHDP (ϵ=10) result has a RMSE of 0.40, the PHDP (ϵ=1) has a

RMSE of 0.37, the dK-2 result has a RMSE of 2.50 and the E-R result has a RMSE of 2.30. Compared to the original Facebook data, the RMSEs are 1.58 of PHDP (ϵ=10), 2.43 of PHDP (ϵ=1), 7.57 of dK-2 (ϵ=10) and 2.13 of E-R.

This experiment suggests that the PHDP anonymized graphs are good at simulating the information broadcasting ability of OSNs. Since the influence maximization problem is closely related to the recommendation and advertisement application, the PHDP anonymized graph achieves high utility.

VI. RELATED WORK

Persistent homology is a description of topology [24]. It has many applications, e.g., analyzing persistent aircraft networks [19], calculating the distance between networks [14], and scheduling robot paths in uncertain environments [2]. Persistent homology is novel in security analysis. Speranzon and Bopardikar achieved K-anonymity [23] based on the zigzag persistent homology [4, 21].

Ghrist proposed the barcode to demonstrate persistent homology [11]. It was applied to analyze the structure of the complex network [13] and random complexes [1]. In our previous research, persistent homology barcodes are introduced to evaluate the utility preservation of existing OSN anonymization schemes. However, none of existing anonymization scheme can preserve persistent structures.

Several mechanisms were employed to anonymize the identity and the network topology. K-anonymity based mechanisms were developed to hide sensitive data among relational data [23]. They were designed for specific parameters, while the adversary can exploit other structural information to deanonymize the data [16]. Fortunately, the differential privacy-based mechanisms were studied to solve these vulnerabilities [7, 10]. Chen et al. employed the adjacency matrix model to achieve differential privacy, which inspires our work because it has the minimum sensitivity values [5].

VII. CONCLUSION

In this paper, we address the utility concerns of the published graph by designing a novel anonymization scheme called PHDP under differential privacy. Unlike the existing anonymization schemes based on traditional components, e.g., node degree or clusters, PHDP employs a novel metric called persistent homology. When the persistent structures are in the form of holes, PHDP preserves the holes as well as satisfies the differential-privacy criteria. Evaluations on real OSNs confirm that protecting the holes help PHDP outperform the other schemes in both the graph utility metric and application metric. In the future, in addition to MCMC as the approximation method, we will try other methods to optimize the noise injection phase.

REFERENCES

[1] Adler, R. J., Bobrowski, O., Borman, M. S., Subag, E., & Weinberger, S. (2010). Persistent homology for random fields and complexes. In *Borrowing Strength: Theory Powering Applications – A Festschrift for Lawrence D. Brown* (pp. 124–143). Institute of Mathematical Statistics. https://doi.org/10.1214/10-IMSCOLL609

[2] Bhattacharya, S., Ghrist, R., & Kumar, V. (2015). Persistent Homology for Path Planning in Uncertain Environments. *IEEE Transactions on Robotics*, *31*(3), 578–590. https://doi.org/10.1109/TRO.2015.2412051

[3] Carlsson, E., Carlsson, G., & De Silva, V. (2006). An algebraic topological method for feature identification. *International Journal of Computational Geometry & Applications*, *16*(04), 291–314. https://doi.org/10.1142/S021819590600204X

[4] Carlsson, G., de Silva, V., & Morozov, D. (2009). Zigzag persistent homology and real-valued functions. *Proceedings of the Twenty-Fifth Annual Symposium on Computational Geometry*, 247–256. https://doi.org/10.1145/1542362.1542408

[5] Chen, R., Fung, B. C. M., Yu, P. S., & Desai, B. C. (2014). Correlated network data publication via differential privacy. *The VLDB Journal*, *23*(4), 653–676. https://doi.org/10.1007/s00778-013-0344-8

[6] Chen, W., Wang, Y., & Yang, S. (2009). Efficient influence maximization in social networks. *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 199–208. https://doi.org/10.1145/1557019.1557047

[7] Cynthia Dwork. Differential privacy. In *Encyclopedia of Cryptography and Security*, pages 338–340. Springer, 2011.

[8] Erdős, P., & Rényi, A. (n.d.). *ON THE EVOLUTION OF RANDOM GRAPHS by*. 45.

[9] Gao, T., & Li, F. (2018). Studying the utility preservation in social network anonymization via persistent homology. *Computers & Security*, *77*, 49–64. https://doi.org/10.1016/j.cose.2018.04.003

[10] Gao, T., Li, F., Chen, Y., & Zou, X. (2018). Local Differential Privately Anonymizing Online Social Networks Under HRG-Based Model. *IEEE Transactions on Computational Social Systems*, *5*(4), 1009–1020. https://doi.org/10.1109/TCSS.2018.2877045

[11] Ghrist, R. (2008). Barcodes: The persistent topology of data. *Bulletin of the American Mathematical Society*, *45*(1), 61–75. https://doi.org/10.1090/S0273-0979-07-01191-3

[12] Gilks, W. R. (2005). Markov chain monte carlo. Encyclopedia of Biostatistics. *Advance online publication. doi*, *10*(0470011815), b2a14021.

[13] Horak, D., Maletić, S., & Rajković, M. (2009). Persistent homology of complex networks. *Journal of Statistical Mechanics: Theory and Experiment*, *2009*(03), P03034. https://doi.org/10.1088/1742-5468/2009/03/P03034

[14] Horak, D., Maletić, S., & Rajković, M. (2009). Persistent homology of complex networks. *Journal of Statistical Mechanics: Theory and Experiment*, *2009*(03), P03034. https://doi.org/10.1088/1742-5468/2009/03/P03034

[15] Ivanov, S., & Karras, P. (2016). Harvester: Influence Optimization in Symmetric Interaction Networks. *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 61–70. https://doi.org/10.1109/DSAA.2016.95

[16] Ji, S., Li, W., Mittal, P., Hu, X., & Beyah, R. (2015). *SecGraph: A Uniform and Open-source Evaluation System for Graph Data Anonymization and De-anonymization*. 303–318. https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/ji

[17] Leskovec, J., & Krevl, A. (2014). SNAP Datasets: Stanford large network dataset collection.

[18] Mischaikow, K., & Nanda, V. (2013). Morse Theory for Filtrations and Efficient Computation of Persistent Homology. *Discrete & Computational Geometry*, *50*(2), 330–353. https://doi.org/10.1007/s00454-013-9529-6

[19] Petri, G., Scolamiero, M., Donato, I., & Vaccarino, F. (2013). Topological Strata of Weighted Complex Networks. *PLOS ONE*, *8*(6), e66506. https://doi.org/10.1371/journal.pone.0066506

[20] Sala, A., Zhao, X., Wilson, C., Zheng, H., & Zhao, B. Y. (2011). Sharing graphs using differentially private graph models. *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, 81–98. https://doi.org/10.1145/2068816.2068825

[21] Speranzon, A., & Bopardikar, S. D. (2016). An algebraic topological perspective to privacy. *2016 American Control Conference (ACC)*, 2086–2091. https://doi.org/10.1109/ACC.2016.7525226

[22] Xiao, Q., Chen, R., & Tan, K.-L. (2014). Differentially private network data release via structural inference. *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 911–920. https://doi.org/10.1145/2623330.2623642

[23] Zhou, B., & Pei, J. (2008). Preserving Privacy in Social Networks Against Neighborhood Attacks. *2008 IEEE 24th International Conference on Data Engineering*, 506–515. https://doi.org/10.1109/ICDE.2008.4497459

[24] Zomorodian, A., & Carlsson, G. (2005). Computing Persistent Homology. *Discrete & Computational Geometry*, *33*(2), 249–274. https://doi.org/10.1007/s00454-004-1146-y