Social Networks Analysis and Mining manuscript No.
(will be inserted by the editor)

Complexity of Social Network Anonymization

Sean Chester · Bruce M. Kapron ·
Gautam Srivastava · S. Venkatesh

Received: date / Accepted: date

Abstract With an abundance of social network data being released, the need to protect sensitive information within these networks has become an important concern of data publishers. To achieve this objective, various notions of k-anonymization have been proposed for social network graphs. In this paper we focus on the complexity of optimization problems that arise from trying to anonymize graphs, establishing that optimally k-anonymizing the label sequences of edge-labeled graphs is intractable. We show how this result implies intractability for other notions of k-anonymization in literature.

We also consider the case of bipartite social network graphs which arise from the representation of distinct entities, such as movies and viewers, patients and drugs, or products and customers. Within this setting we demonstrate that, although k-anonymizing edge-labeled graphs is intractable for $k \geq 3$, polynomial time algorithms exist for arbitrary bipartite graphs when $k = 2$ and for unlabeled bipartite graphs irrespective of the value of k.

Finally, in this paper we extend the study of attribute disclosure within the context of social networks by defining t-closeness, a measure of how effectively an adversary can determine sensitive information about members of a k-anonymous social network.

Keywords privacy · social networks · complexity · k-anonymity · table graphs

1 Introduction

The social web's recent explosive growth is exciting for the novel analysis and mining opportunities it presents. Entities and the links between them emerge

S. Chester, B. M. Kapron, G. Srivastava, S. Venkatesh
PO Box 3055 STN CSC
Victoria BC Canada V8W 3P6
E-mail: schester@uvic.ca, bmkapron@cs.uvic.ca, gsrivast@uvic.ca, venkat@cs.uvic.ca

on countless platforms and out of myriad websites, providing social data at an unprecedented level, the characteristics of which can be quite unique to context (see, for example, the study by Cha et al. [7] of content dissemintation in blogs). An especially interesting example is the PatientsLikeMe social network platform.[1] Here, members (entities) get the chance to connect (establish links) with others who are dealing with similar health issues. Measuring the strength of associations that formed through the platform could provide vital data in the study for disease research. However, the social network platform inherently contains sensitive information that patients would not want divulged; can we ensure that the meaningful study of such platforms will not compromise the confidentiality of their participants?

It has already been shown that naive attempts to hide this sensitive information do not work [4,13]. In the last five years, more sophisticated *k-anonymization*-based techniques have been proposed by adopting the idea that the confidentiality of the participants can be protected if the structural properties used for an attack are not unique to fewer than k participants. Each technique differs in terms of the structural property under consideration. What is not apparent from this literature, however, is the computational challenge involved in establishing the k-anonymous conditions without severly distorting the network.

In this paper, we undertake a systematic investigation of the hardness of anonymization, establishing not only previously unknown hardness results for many notions of graph k-anonymization but also a framework for determining hardness of future notions of graph k-anonymity. We accomplish this via two natural generalizations of the k-degree anonymity present in literature. First, when social network data is represented as a graph, we would perhaps prefer to anonymize only a subset of the nodes. For example, in a social network, some users may agree to have their information released, while others wish to remain anonymous (see Yuan et al. [23]). This generalization gives rise to what we introduce and focus on, the problem of *subset anonymization*. Second, the graph representing a social network often has labeled edges. A label provides auxiliary information regarding an association. As an example, one can construct an implicit social network among shoppers by linking them to products they have purchased, and the label on each edge could be data such as dates of purchase, quantities, ratings, etc. In general, edge weights allow one more sophistication in modeling the network, which could lead to more sophisticated analysis (such as with the generalized network measures proposed by Adbdallah [1]).

These considerations lead to our *k-label sequence subset anonymity* problem in which we are given an edge-labeled graph G and we would like to ensure that a given subset of vertices of G is *k-label sequence anonymous* by adding a minimum number of edges. We will also study this problem for bipartite graphs, where the vertices to be anonymized are from one side of the bipartition. The bipartite model is useful in cases where vertices represent two types

[1] `http://www.patientslikeme.com`.

of entities, edges exist only between entities of different types, and only one type of entity needs to be anonymized. An example of this would be a graph representing interactions between customers and products. (See the study of bipartite network graphs by Zweig and Kaufmann [26] for other examples.)

Finally, we look beyond the issue of identity disclosure in social networks to the privacy concern of attribute disclosure. We adapt for graphs the notion of *t-closeness* from the table anonymity literature and prove that it, too, is NP-complete when coupled with k-anonymization based on k-vertex-label sequence anonymity.

Our Results We introduce algorithms and hardness results for labeled and unlabeled graphs. In the edge-labeled case we consider k-anonymization with respect to the collection of labels of incident edges. In §5 we deal with k-anonymization of subsets in arbitrary labeled graphs. Using a class of graphs, *table graphs*, that we introduce in §4, we prove the hardness of many seemingly different notions of graph anonymization that have already appeared in the literature, providing a uniform approach to the complexity of graph anonymization problems.

In §6 we consider subset k-anonymization of bipartite graphs. Considering first labeled graphs, we provide, for $k = 2$, a polynomial time algorithm based on a recent algorithm of Anshelevich and Karagiozava [3] for finding minimum weight perfect matching in hypergraphs with edges of size two or three. When $k \geq 3$ we show that the problem is NP-complete.

In the unlabeled case, we consider k-anonymization with respect to the degree of vertices. In §6.2 we present an algorithm for subset k-degree anonymization of unlabeled bipartite graphs that runs in time $O(n(k + d_{max}) + n \log n)$, where n is the number of vertices in the graph and d_{max} is the maximum degree of a vertex in the graph. We use a dynamic programming approach to achieve this bound.

Finally, in §7, we motivate adapting from the table privacy literature the notion of t-closeness (Definition 12), a more robust preventive measure for attribute disclosure attacks than yet published. We show that if this is considered in conjunction with a natural choice of preventitive measure for identity disclosure attacks, then it is NP-complete, too.

2 Related Work

In recent years, many interesting definitions for graph anonymization have been proposed and studied. Each of them starts by modeling the background information that an adversary will use to attack the data. Once that is done, a notion of anonymity is defined and studied.

Liu and Terzi proposed a simple graph anonymization technique to prevent identity disclosure attacks [16]. They assume that the adversary has prior knowledge of degrees of certain vertices in the network, and may use this information to try and identify certain nodes in the network. To fight such attacks, they defined the concept of k-degree anonymity. A graph $\mathrm{G} = (\mathrm{V}, \mathrm{E})$ is said to be k-degree anonymous if for every vertex $v \in \mathrm{V}$, there are at least $k - 1$ other vertices in V with equal degree to that of v.

Hay et al. [13] model the information available to the adversary using two types of queries–vertex refinement queries and subgraph knowledge queries–and study the vulnerability of various datasets under such an attack. They propose an anonymization technique based on random perturbations against such adversaries.

Zheleva and Getoor [24] study the problem of protecting certain sensitive edges in an edge-labeled graph under link re-dentification attacks. They propose anonymization techniques using edge-removal and node-merging to prevent such attacks.

Zhou and Pei [25] focus on neighbourhood attacks, which was expanded by Tripathy and Panda [21]. In their model, an adversary uses information about a node's neighbours to target them. To prevent such attacks, they define a notion of k-anonymity on graphs so that nodes in an anonymized group will have isomorphic neighbourhoods.

Thompson and Yao [20] study i-hop degree-based attacks. In their model an adversary's prior knowledge includes the degree of the target and the degree of its neighbours within i hops. Thomson and Yao use bipartite graphs, namely the Netflix Prize Data, to help motivate their work.

Wu et al. [22] proposed the k-symmetry model. They state for any vertex v in the network, there exists at least $k-1$ structurally equivalent counterparts. The authors also proposed sampling methods to extract approximate versions of the original network from the anonymized network so that statistical properties of the original network could be evaluated.

Cormode et al. [10] consider a new family of anonymizations for bipartite graph data called (k, l)-groupings. These groupings were used to preserve the underlying graph structure perfectly, and instead anonymize the mapping from entities to nodes of the graph. They created "safe" groupings that were able to withstand a set of known attacks.

A common limitation among most of this work is that there is no study of the hardness of the privacy notions proposed. An earlier version of our work appeared at ASONAM 2011 [14]; combined with our further contributions here, this is the first research to address this limitation.

On another note, Li et al. [15] identify two types of privacy attack for data, namely identity disclosure and attribute disclosure. Identity disclosure often leads to attribute disclosure. Identity disclosure occurs when an individual is identified within a dataset, whereas attribute disclosure occurs when sensitive information that an individual wished to keep private is identified. These aforementioned works all protect against identity disclosure.

Regarding attribute disclosure, Machanavajjhala et al. [17] introduced for tabular data the notion of l-diversity, wherein each k-anonymous equivalence class requires l different values for each sensitive attribute. In this way, l-diversity looks to not only protect identity disclosure, but was also the first attempt to protect against attribute disclosure. Zhou and Pei adapt the work of Machanavajjhala et al. by defining l-diversity for graphs [25].

To address the shortcomings of l-diversity, Li et al. [15] introduced t-closeness, which requires that the distribution of attribute values within each k-anonymous equivalence class needs to be close to that of the attribute's distribution throughout the entire table. This work has not been adapted for the social network setting, which is a contribution of ours in §3. We also note that a form of attribute disclosure has been studied in our earlier work [9], but the work is dissimilar in spirit to this in that we strove there to protect the attribute of a target vertex's friends, as opposed to that of the target itself.

3 Preliminaries

In this section, we define the concept of k-anonymity for tables, unlabeled graphs and labeled graphs. While notions of k-anonymity for tables and unlabeled graphs have been studied previously, k-anonymity for labeled graphs is introduced in this paper.

Throughout, we investigate numerous subset anonymization problems. Here we abstractly describe the primary problem of interest throughout the paper:

Problem 1 (k-Subset Anonymization Problem (k-SAP):) *Given a graph* $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \Sigma)$*,* $X \subseteq \mathrm{V}$*, find a graph* $\mathrm{G}' = (\mathrm{V}, \mathrm{E} \cup \mathrm{E}', \Sigma \cup \Sigma')$ *such that* $\mathrm{E}' \subseteq \mathrm{V} \times \mathrm{V} \times (\Sigma \cup \Sigma')$*, the sequence corresponding to* X *is* k*-anonymous in* G'*, and the number of new edges added,* $|\mathrm{E}'|$*, is minimized.*

3.1 Tables and k-Anonymity

Table Anonymization has been extensively studied [2,6,11,12,18,19]. Suppose we want to publish a table of data containing potentially sensitive information. Each attribute in the table can be considered as either an identifying attribute (such as social insurance number, or student id), a quasi-identifier (such as age or postal code) which combined with other quasi-identifiers can reveal the identity of a record, or a sensitive attribute (such as disease or income).

Clearly, identifying attributes must be stripped from the table, but this alone does not guarantee privacy. To help protect the data, we have the ability to suppress the data entries in the quasi-identifier attributes of the table with *'s. To achieve k-anonymization by suppressing the entries, we require that after suppression, for any given row in the table, there are $k-1$ other rows that look identical with respect to the quasi-identifying attributes. Throughout this paper, we treat tables as having only quasi-identifying attributes, because the sensitive attributes do not affect k-anonymization.

If we want to 2-anonymize the example table data in Figure 1(a), then using the fewest suppressions to achieve 2-anonymity would produce the table in Figure 1(b).

Fname	LName	Age	Grad Year
Harry	Potter	30	2012
John	Connor	45	2013
Harry	Houdini	30	2010
Sarah	Connor	32	2013

(a) Example table data prior to anonymization

Fname	LName	Age	Grad Year
Harry	*	30	*
*	Connor	*	2013
Harry	*	30	*
*	Connor	*	2013

(b) Optimal 2-anonymization of example table data

Fig. 1 Example of k-anonymization of table data

Definition 1 A table consists of a multiset V of *rows*, that is, sequences of length m over a set Σ of *entry values*. Let $t : V \longrightarrow (\Sigma \bigcup \{*\})^m$. If for all $v \in V$ and $j = 1, \ldots, m$ it is the case that $t(v)_j \in \{v_j, *\}$, we call t a *suppressor*. The table $t(V)$ resulting from a suppressor t is defined to be k-anonymous iff for all $v \in V$ there exist at least $k-1$ distinct rows $v_1, \ldots, v_{k-1}$ such that $t(v) = t(v_1) = \ldots = t(v_{k-1})$. In other words, after applying t, each row is identical to at least $k-1$ other rows.

3.1.1 Anonymizing entries is hard

Meyerson and Williams [18] showed that the problem of finding the minimum number of suppressions to anonymize a table was proven NP-hard for $k \geq 3$ and $|\Sigma| \geq n$, where n denotes the number of rows of the table ($|V|$). From this, Aggarwal et al. [2] lowered the alphabet size to $|\Sigma| = 3$. Finally, it was shown by Bonizzoni et al. [6] that the problem remains hard for $|\Sigma| = 2$ and $k \geq 3$.

3.2 Unlabeled Graphs and k-Anonymity

Let $\mathrm{G} = (\mathrm{V}, \mathrm{E})$ be a simple graph where V denotes the set of vertices and E denotes the set of edges. We denote the degree of a vertex v by $d(v)$. The analogous notion of k-anonymity in the social network setting is to exploit structural knowledge of the graph as a quasi-identifier. In this way, vertices in the social network graph must be identical to at least k-1 other vertices with respect to that structural knowledge. Here we provide the definitions needed to formalise the description of an attack. In general anonymizing graphs to prevent structural attacks is quite challenging, because, unlike in the table setting, vertices are not independent of each other: altering the structure of one vertex necessitates altering the structure of another simultaneously.

Definition 2 (Degree Sequence) Let $X = \{x_1, x_2, \ldots, x_n\}$, $X \subseteq \mathrm{V}$, be a subset of vertices of G. The *degree sequence* of X is $(d_1, d_2, \ldots, d_n)$ where $d_i = d(x_i)$ is the degree of the vertex x_i.

Definition 3 (Degree Anonymity) A sequence of values $S = (s_1, s_2, \ldots, s_n)$ is said to be *k-degree-anonymous* if every distinct value in S occurs at least k times. A subset of vertices X in an unlabeled graph G is k-degree-anonymous if its degree sequence is k-degree-anonymous.

Problem 2 (k-Degree-Based Subset Anonymization Problem (k-D-SAP)): *Given a graph* $\mathrm{G} = (\mathrm{V}, \mathrm{E})$*,* $X \subseteq \mathrm{V}$*, find a graph* $\mathrm{G}' = (\mathrm{V}, \mathrm{E} \cup \mathrm{E}')$ *such that X is k-degree-anonymous in* G' *and the number of new edges added,* $|\mathrm{E}'|$*, is minimized.*

Note: We state our anonymization problems in the optimization version of [2,6,18], and indeed the algorithms we give are naturally viewed in this way. On the other hand, for hardness, we in fact deal with the decision version of these problems. That is, we have another input $t \in \mathbf{N}$, and we ask whether there is a set E' of edges such that G' is k-anonymous and $|\mathrm{E}'| \leq t$.

Example 1 Here we present a small example of **k-D-SAP**. Consider the graph G in Figure 2(a). Suppose we want 2-anonymity for the subset of vertices $\{v_1, v_2, v_5, v_6\}$, which has degree sequence $(2, 4, 2, 2)$. Adding the dotted edges of Figure 2(b) will result in the degree sequence $(2, 4, 2, 4)$, which is 2-anonymous. Since, for 2-anonymity, we require at least 2 vertices of degree 4 in the sequence, the number of edges added is the minimum.

3.3 Labeled Graphs and k-Anonymity

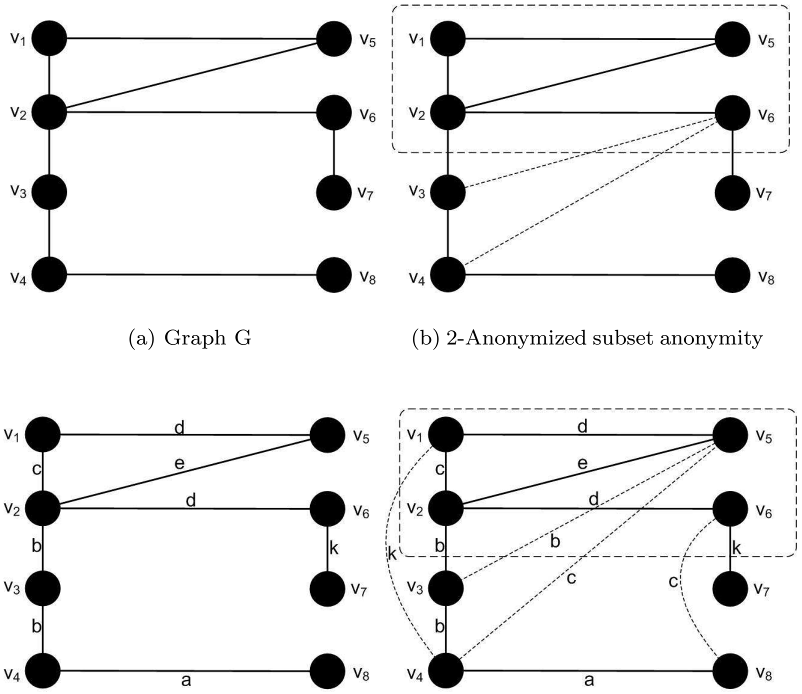
Edge-labeled graphs are a natural model for the representation of social networks and related forms of data. The PatientsLikeMe database, for example, can be represented with nodes users and labeled edges to represent how strongly or frequently users interact with each other.

Definition 4 (Edge-labeled Graph) An *edge-labeled graph* is a tuple $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \Sigma)$ where V is the set of *vertices*, Σ is the *label set* and $\mathrm{E} \subseteq \mathcal{P}_2(\mathrm{V}) \times \Sigma$, is the set of *(labeled) edges*. Here $\mathcal{P}_2(\mathrm{V})$ denotes the 2-element subsets of V. E must satisfy the property that there is at most one $\ell \in \Sigma$ such that $(\{u, v\}, \ell) \in \mathrm{E}$. If $(\{u, v\}, \ell) \in \mathrm{E}$ is a labeled edge, we say that ℓ is the *label* of edge $\{u, v\}$.

Definition 5 (Label Sequence) For $v \in \mathrm{V}$, we say that $S_v = (\ell_1, \ell_2, \ldots, \ell_m)$ is a *label sequence* of v if it corresponds to some ordering of the labels of the edges incident on v. We consider label sequences to be equivalent up to permutations.[2]

Definition 6 (Label Sequence Anonymity) Given an edge-labeled graph $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \Sigma)$, a subset $X \subseteq \mathrm{V}$ of vertices is *k-label sequence anonymous in* G if for every vertex v in X, there are at least $k-1$ vertices in X whose label sequence is equivalent to the label sequence of v. If v and v' are vertices with equivalent label sequences we say that they are *similar* and write $v \equiv v'$.

[2] We use permutation-invariant sequences rather than multisets to avoid the need to deal explicitly with multiplicities.



(a) Graph G (b) 2-Anonymized subset anonymity

(c) Graph H (d) 2-Label sequence anonymization of X in H

Fig. 2 Example 1: k-**D-SAP** and Example 2: k-**LS-SAP**

Clearly $\equiv$ is an equivalence relation and so induces a partition $X/\!\equiv$ of X. We now define the anonymization problem for subsets of labeled graphs.

Problem 3 (k-Label Sequence-Based Subset Anonymization Problem (k-LS-SAP)): *Given an edge-labeled graph* $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \Sigma)$*,* $X \subseteq \mathrm{V}$*, find an edge-labeled graph* $\mathrm{G}' = (\mathrm{V}, \mathrm{E} \cup \mathrm{E}', \Sigma \cup \Sigma')$ *such that* X *is* k*-anonymous in* G' *and the number of edges added,* $|\mathrm{E}'|$*, is minimized. We call* E' *an* anonymizing set of edges *for* X*.*

In other words, we would like to k-anonymize X by adding the minimum number of new labeled edges to G. Note that the added edges may have labels from an expanded set $\Sigma \cup \Sigma'$.

Example 2 Here we present an example of subset label sequence anonymization. Consider graph H in Figure 2(c). Here, if we have $X = \{v_1, v_2, v_5, v_6\}$, with $k = 2$, similar to Example 1, adding the dotted edges in Figure 2(d) with the given edge labels gives us 2-label-sequence-based anonymity. In this case it is not sufficient just to have a 2-anonymous degree sequence; we must also consider the labels of incident edges for each vertex.

Patient	Ailment
A	Flu
B	Fever
C	ADD
D	Cancer
E	Heart Disease

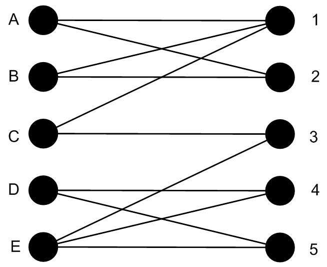
(a) Patient table

Drug	Name
1	Ritalin
2	Aspirin
3	Tylenol
4	Penicillin
5	Diuretic

(b) Drug table

Pat	Drug	Pat	Drug
A	1	D	4
A	2	D	5
B	1	E	3
B	2	E	4
C	1	E	5
C	3		

(c) Patient-Drug join



(d) $G = (V, W, E \subseteq V \times W)$

Fig. 3 A join relation between patients and drugs and the corresponding bipartite graph

3.4 Tables as Bipartite Graphs

As previously mentioned, table data is often easily represented using graphs, particularly bipartite graphs.

Definition 7 A *simple bipartite graph* is a triple (V, W, E) where V and W are *vertex sets*, and $E \subseteq V \times W$ is the set of *edges*. The pair (V, W) is called a *bipartition*, and V and W are respectively called the *left* and *right sides* of the bipartition.

Example 3 Consider a relational database consisting of a table for patients, a table for prescription drugs, and a table for the treatment of patients with the drugs. In Figure 3, we see an example instance of this database, and also its representation using a bipartite graph. Here patients are represented by vertices in V, drugs by vertices in W, and the viewing relation is represented by edges between V and W. We could label edges in the graph of Figure 3(d) to introduce more information such as treatment time for each patient with the given drug.

3.5 Vertex-labeled Graphs and t-Closeness

Another type of labeled graph is a *vertex-labeled* graph, in which labels are associated with vertices rather than edges. Such graphs arise naturally out of a social network context when the supplementary information embedded in the network relates to entities rather than associations. For example, we could label the left vertices of the graph in Figure 3(d) with their ailment indicated in Figure 3(a). In this case, the vertex contains a *sensitive* label, because it has been labeled with a sensitive attribute from the table. We do note that a vertex-labeled graph is a special case of an edge-labeled graph.[3] Nonetheless, they are very interesting in their own right because they arise naturally and because they have implications in terms of attribute disclosure when the label is sensitive.

Before detailing our notion of anonymity to protect against attribute disclosure, we formalize the attack. Specifically, we are assuming that the social network is an undirected, simple, vertex-labeled graph:

Definition 8 (Vertex-labeled Graph) A *vertex-labeled graph* is a graph $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \Sigma, \ell)$, where V is a vertex set, $\mathrm{E} \subseteq \mathrm{V} \times \mathrm{V}$ is the edge set, Σ is an alphabet of labels, and $\ell = \mathrm{V} \mapsto \Sigma$ is a labeling function that assigns a label $l \in \Sigma$ to each vertex in V For simplicity, $(u, v) \in \mathrm{E} \rightarrow (v, u) \in \mathrm{E}$. We assume for convenience that the elements of Σ are ordered.

An attribute disclosure attack occurs when an adversary can refine his knowledge about the label of a target vertex. To model this knowledge gain, it is important to consider the distribution of labels over a set of vertices:

Definition 9 (Label Distribution) For $\mathrm{W} \subseteq \mathrm{V}$, let $\mathrm{count}(l_i, \mathrm{W})$ denote the number of vertices in the set W which have label l_i. Then, the *label distribution* over W, denoted distr(W), is the vector $\frac{\langle \mathrm{count}(l_1,\mathrm{W}),\ldots,\mathrm{count}(l_{|\Sigma|},\mathrm{W})\rangle}{|\mathrm{W}|}$. Also, the distribution of a particular label, l_i is one element of the vector, $\mathrm{count}(l_i, \mathrm{W})/|\mathrm{W}|$.

Then, given two distributions $\mathrm{distr}(\mathrm{W}_1), \mathrm{distr}(\mathrm{W}_2)$, $\mathrm{W}_1, \mathrm{W}_2 \subseteq \mathrm{V}$ we define a distance measure as the standard L_1 norm:

Definition 10 (Distance Δ between Two Distributions) Given two distributions, $\mathrm{distr}(\mathrm{W}_1) = \left\langle \mathrm{W}_{11}, \ldots, \mathrm{W}_{1|\Sigma|} \right\rangle$ and $\mathrm{distr}(\mathrm{W}_2) = \left\langle \mathrm{W}_{21}, \ldots, \mathrm{W}_{2|\Sigma|} \right\rangle$, the distance between them, denoted $\Delta(\mathrm{distr}(\mathrm{W}_1), \mathrm{distr}(\mathrm{W}_2))$, is

$$\Delta(\mathrm{distr}(\mathrm{W}_1), \mathrm{distr}(\mathrm{W}_2)) = \sum_{i=1}^{|\Sigma|} |\mathrm{W}_{1i} - \mathrm{W}_{2i}|.$$

For example, the distributions $\langle .7, .2, .1 \rangle$ and $\langle .2, .4, .4 \rangle$ have a distance of .5+.2+.1=.8.

[3] To see this, consider that any vertex-labeled graph can be transformed into a unique edge-labeled graph by labeling every edge (u, v) as $\{l(u), l(v)\}$.

Given these definitions, we can now formally describe the adversary's attack. By identifying the equivalence class Y of a node v in a vertex-labeled graph, an adversary gains knowledge about the sensitive label of v. Whereas beforehand, he could only surmise the probability that that label of v is l_i to be $\mathrm{count}(l_i, \mathrm{V})/|\mathrm{V}|$, he now knows that the probability is closer to $\mathrm{count}(l_i, Y)/|Y|$. That is:

Definition 11 (Attribute Disclosure (AD) Attack) An *attribute disclosure attack* against a vertex v in a graph $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \Sigma, \ell)$ is one in which the adversary, by knowing the equivalence class $Y \subseteq \mathrm{V}$ containing v, discovers a more refined estimate of the label of v than he had when he only knew $\mathrm{distr}(\mathrm{V})$. His knowledge gain is $\Delta(\mathrm{distr}(\mathrm{V}), \mathrm{distr}(Y))$.

To combat this type of attack, we define *t-closeness* for graphs, wherein the distance from the label distribution of any equivalence class to that over the entire vertex set must be within t:

Definition 12 (t-Closeness) An equivalence class Y is said to be *t-close* in a graph $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \Sigma, \ell)$ if $\Delta(\mathrm{distr}(Y), \mathrm{distr}(\mathrm{V})) \leq t$. A vertex-labeled graph G is *t-close* if every equivalence class of V is itself t-close.

We leave the definition of equivalence class of V open, so that t-closeness is defined in conjunction with any notion of k-anonymity, whether based on label sequence, isomorphism, or symmetry. We consider it in conjunction with label sequences in §7.

4 A Unified Framework for Establishing Hardness for *k*-Anonymization Problems

In this section, we introduce a special class of graphs called *table graphs*. Our new notion of *table graphs* presented here can be viewed as a unifying framework to prove hardness results for graph k-anonymization. Many earlier papers showed schemes that worked well in practice. However, the complexity of the various notions of graph anonymization are poorly understood (with the exception of Zhou and Pei [25] who showed the hardness of neighbourhood anonymity for vertex-labeled graphs). We demonstrate here that many of these various notions (e.g., neighbourhood anonymity (§5.2), 1-hop anonymity (§5.3), and k-symmetry anonymity (§5.4)) can be reduced to k-**Table Graph Anonymization** in order to establish their hardness.

Definition 13 (Table Graphs) An edge-labeled graph $\mathrm{G} = ((\mathrm{U}, \mathrm{V}, \mathrm{W}), \mathrm{E}, \Sigma)$ is an $n \times l$ table graph if:

- $|\mathrm{U}| = n$ and $|\mathrm{V}|, |\mathrm{W}| = l$ for some n and l
- $\mathrm{E} \subseteq (\mathrm{U} \times \mathrm{V} \times \Sigma) \cup (\mathrm{U} \times \mathrm{W} \times \Sigma)$
- All edges incident to $v_i \in \mathrm{V}$, $1 \leq i \leq l$, are labeled $2(i-1)$
- All edges incident to $w_i \in \mathrm{W}$, $1 \leq i \leq l$, are labeled $2(i-1)+1$.

Problem 4 (k-Table Graph Anonymization):
Given an $n \times l$ table graph $\mathrm{G} = ((\mathrm{U}, \mathrm{V}, \mathrm{W}), \mathrm{E}, \Sigma)$ and $X \subseteq \mathrm{W}$, construct an $n \times l$ table graph $\mathrm{G}' = ((\mathrm{U}, \mathrm{V}, \mathrm{W}), \mathrm{E} \cup \mathrm{E}', \Sigma \cup \Sigma')$ such that X is k-label sequence anonymous in G' and $|\mathrm{E}'|$ is minimized.

In the next section we prove that k-**LS-SAP** is NP-complete, a corollay of which is that k-**Table Graph Anonymization** is NP-complete. We then illustrate the application of our unifying framework to establishing hardness for other measures of graph anonymization, namely neighbourhood anonymization, k-symmetry, and i-hop anonymity. It is easily verified that a polynomial-size certificate of membership in NP exists for all three problems; therefore, we are demonstrating that the problems are NP-complete.

5 Hardness of LS-SAP and Table Graph Problems

In this section, we prove Theorem 1. We then use this result to show NP-completeness of many different notions of graph anonymization introduced recently.

Theorem 1 *For $k \geq 3$, k-**LS-SAP** is NP-complete.*

5.1 Labeled Sequence and Table Graph Anonymization

Let $k \geq 3$ be any fixed integer. To show hardness of k-**LS-SAP** we build a reduction to the decision version of k-**LS-SAP** from the NP-hard table anonymization problem introduced in §3.

Problem 5 (k-ENTRY-ANONYMITY):
Input: *a table T with n rows and l columns (also called attributes) with entries over $\{0, 1\}$ and an integer t.*
Question: *Can the rows of T be k-anonymized by suppressing at most t entries of T? Here, an entry (0 or 1) is said to be suppressed if it is replaced by *.*

Reduction: Our reduction is described as follows: given a table T, let $\mathrm{T}_{(m,j)} \in \{0, 1\}$ denote the value of attribute j in row m. Then, the edge-labeled graph $\mathrm{G_T}$ corresponding to T is constructed as follows:

- $\mathrm{V_T} = \{r_1, r_2, \ldots, r_n\} \cup \{c_j^i | 1 \leq j \leq l, i \in \{0, 1\}\}$.
- $\mathrm{E_T} = \{(r_m, c_j^i, 2(j-1)+i) | \mathrm{T}_{(m,j)} = i, 1 \leq m \leq n, 1 \leq j \leq l, i = 0, 1\} \cup \{(r_i, r_j, 2l) | 1 \leq i, j \leq n\}$.
- $\Sigma_\mathrm{T} = \{0, 1, \ldots, 2l\}$.
- Finally, remove all isolated vertices from $\mathrm{G_T}$.

In other words, we encode a binary table as an edge-labeled graph in which a row vertex $r_m \in \mathrm{V_T}$ is connected to a column vertex $c_j^0 \in \mathrm{V_T}$ (alt., $c_j^1 \in \mathrm{V_T}$) with label $2(j-1)$ $(2(j-1)+1)$ if the (m, j)th entry of the table is 0 (1).

Let $X = \{r_1, \ldots, r_n\}$ denote the set of row vertices of $\mathrm{G_T}$. Since there are already edges between every pair of vertices in X, no anonymizing edges will be added between these vertices. We will show that T can be k-anonymized by suppressing at most t entries if and only if we can k-anonymize X by adding at most t new labeled edges.

Let $\mathrm{G'_T}$ be any graph obtained from $\mathrm{G_T}$ such that X is k-anonymous in $\mathrm{G'_T}$ and it has the minimum number of new edges added. Suppose that $\mathrm{E'_T}$ is an anonymizing set of edges for X. Letting $\equiv$ denote vertex similarity in the anonymized graph, let $Y = \{y_1, \ldots, y_m\}$ be an equivalence class of $X/\equiv$, where $m \geq k$. We begin by establishing properties that any anonymizing set $\mathrm{E'_T}$ of minimum size must satisfy.

Let $Y = \{y_1, \ldots, y_m\}$ be an equivalence class of $X/\equiv$, where $m \geq k$. Lemma 1 shows that the anonymization procedure only introduces edges with labels already in Σ_T.

Lemma 1 *If there is an edge in $\mathrm{E'_T}$ labeled ℓ that is incident to Y then there is an edge in $\mathrm{E_T}$ labeled ℓ that is incident to Y.*

Proof Suppose ℓ is the label of an edge in $\mathrm{E_T} \cup \mathrm{E'_T}$ that is incident to a vertex $y \in Y$. Then there must be an edge in $\mathrm{E_T}$ with label ℓ incident to some vertex $y' \in Y$. If this were not the case, then we may remove all edges labeled ℓ from $\mathrm{E'_T}$ which are incident to vertices in Y, and maintain the similarity of all vertices in Y with a smaller anonymizing set of edges.

Lemma 2 shows that at most one edge with label ℓ is incident to a row vertex of $\mathrm{V_T}$.

Lemma 2 *For every $i \in \{0, 1\}$, and every $j \in \{1, 2, \ldots, l\}$, the label $2(j-1)+i$ appears at most once in the label sequence of a vertex $y \in Y$.*

Proof We first show that if there is an edge in $\mathrm{E_T}$ labeled ℓ that is incident to $y \in Y$ then there is no edge in $\mathrm{E'_T}$ labeled ℓ that is incident to y. The proof proceeds by contradiction. Suppose there is such an edge labeled ℓ in $\mathrm{E_T} \cup \mathrm{E'_T}$ that appears more than once. So the label ℓ occurs more than once in the label sequence of y, and hence of every node in Y. By construction only one of these occurrences is due to an edge in $\mathrm{E_T}$. We may remove the edges in $\mathrm{E'_T}$ corresponding to the other occurrences and maintain the similarity of all vertices in Y with a smaller anonymizing set $\mathrm{E'_T}$. On the other hand, suppose that there is an edge labeled ℓ in $\mathrm{E'_T}$ that appears more than once and is incident to y but there is no edge labeled ℓ in $\mathrm{E_T}$. Then, we note again we may remove all edges labeled ℓ from $\mathrm{E'_T}$ which are incident to vertices in Y, and maintain the similarity of all vertices in Y with a smaller anonymizing set $\mathrm{E'_T}$.

Lemma 3 *There is no edge labeled $2l$ in $\mathrm{E'_T}$ that is incident to Y.*

Proof Suppose that there is an edge labeled 2ℓ in E'_T incident to $y \in Y$. Since the number of edges labeled 2ℓ is the same for every vertex of Y in G_T, there must be an edge labeled 2ℓ in E'_T incident to every vertex of Y. We may remove all edges labeled 2ℓ from E'_T which are incident to vertices in Y, and maintain the similarity of all vertices in Y with a smaller anonymizing set of edges.

We now give a proof of correctness of our reduction.

Lemma 4 *Given a Table* T*, the rows of* T *can be made* k*-anonymous by suppressing at most* t *entries if and only if* X *can be made* k*-label sequence anonymous by adding at most* t *edges.*

Proof (If:) By Lemmata 1 and 2, it is clear that for each $y \in Y$ and each j, $1 \leq j \leq l$, y will either

1. Have exactly one incident edge labeled $2(j-1)$ but no incident edge labeled $2(j-1)+1$.
2. Have exactly one incident edge labeled $2(j-1)+1$ but no incident edge labeled $2(j-1)$.
3. Have exactly one incident edge labeled $2(j-1)$ and exactly one incident edge labeled $2(j-1)+1$

This gives us an anonymization of the rows in T corresponding to Y. Namely, in cases (1) or (2) we leave the corresponding table entry unchanged. In case (3) we put a $*$ in the corresponding entry. Note that the number of times that (3) occurs is exactly the number of edges in E'_T incident to Y. We repeat this for each equivalence class in $X/\equiv$, and so conclude that if G can be k-anonymized by adding edges E'_T, then T can be k-anonymized by the suppression (i.e. replacement by a $*$) of $|\mathrm{E}'_\mathrm{T}|$ entries.
(Only if:) Going from an anonymized table to an anonymized graph is quite simple. If the anonymization procedure puts a * in place of value i in entry (m, j) of table T, the graph anonymization procedure we will add an edge from x_m to $c_j^{(1-i)}$ with weight $2(j-1)+(1-i)$. If T is properly anonymized, each row m will have $k-1$ rows that are identical to it. But then in G'_T, vertex x_m will be similar to the vertices corresponding to those $k-1$ rows. Intuitively, we may view the suppression of an entry as putting both a 0 and 1 value in that entry, and adding the corresponding edges to the graph.

Thus, we have the main theorem and corollary of this section:

Theorem 1 *For* $k \geq 3$*,* k***-LS-SAP*** *is NP-complete.*

Proof From Lemmata 1-4 we have that the decision version of this problem can be reduced from k**-ENTRY-ANONYMITY**. Also, the decision version of this problem is in NP. To show this, we note that the collection of the new edges to be added along with the resulting partition of X into k-anonymized subsets is a polynomial-size certificate of membership. Therefore, k**-LS-SAP** is NP-complete.

Corollary 1 *k-**Table Graph Anonymization** is NP-complete.*

We finish this subsection with an observation that will be useful for us later. It uses the idea that for any edge between a row vertex and an attribute vertex, we can always change the attribute vertex endpoint as it does not affect the label sequence of the row vertex.

Lemma 5 *We can assume, without loss of generality, that in G'_T all edges with label $2(j-1)+i$ are only of the form (r_k, c^i_j) for some k, $1 \leq k \leq n$.*

Proof Given an anonymized graph G'_T, one can move edges to their proper location in G'_T and not affect the anonymity. Notice that the anonymization is based of the labels on the edges, not their endpoint vertices, so moving the edges such that they follow the structure of the original graph G_T makes no change to the anonymous label sequences or anonymous groups.

5.2 Neighbourhood Anonymization

In neighbourhood anonymization, of interest is the induced graph of the immediate neighbours of a vertex v. Zhou and Pei [25] studied neighbourhood attacks in which the adversary uses prior knowledge of the connectivity of the neighbours of a target node in a social network for identity disclosure. While Zhou and Pei studied this notion for vertex labeled graphs and proved NP-hardness, we prove here that the problem is also hard for edge-labeled graphs. Neighbourhood anonymity is defined as follows:

Definition 14 (Neighbourhood Anonymity) In an edge-labeled graph $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \varSigma)$, the neighborhood of $u \in \mathrm{V}$ is the induced subgraph on u and the vertices adjacent to u. A graph G is said to be *k-neighbourhood anonymous* if for a given vertex $v \in \mathrm{V}$, there are $k-1$ other vertices in V with a neighborhood isomorphic to that of v.

For this problem, as in the case of ***k*-LS-SAP**, one is given an edge labeled graph G and a subset of vertices X. One needs to add the fewest number of edges to G to make X k-neighbourhood anonymous. We reduce ***k*-Table Graph Anonymization** to k-neighbourhood anonymity.

Lemma 6 *Given a table graph G_T, X can be made k-label sequence anonymous by adding at most j edges if and only if it can be made k-neighbourhood anonymous by adding at most j edges.*

Proof (If:) This is clear since k-neighbourhood anonymity implies k-label sequence anonymity.
(Only if:) By Lemma 5, the optimal anonymization procedure for k-label sequence anonymizing X will result in the same set of neighbours for every y in Y, where Y is an equivalence class of X. Since the set of neighbours is the same, the induced subgraphs on the neighbours are also the same. Hence, for table graphs, k-label sequence anonymity implies k-neighbourhood anonymity. Therefore, k-neighbourhood anonymity is NP-hard.

5.3 1-Hop Anonymization

Thompson and Yao [20] introduced i-hop anonymity, which focuses on the degrees of the immediate neighbours of a node. The assumption is that information about a node may be inferred from information about its immediate neighbours. This assumption is similar to that of Zhou and Pei [25], that if information about a given target node and its immediate neighbours is known to an adversary, the adversary can then use this information to attack the identity of the target node. We will show here that 1-hop labeled subset anonymity is NP-hard. We define i-hop anonymity for edge-labeled graphs as follows.

Definition 15 (i-hop Anonymity) The *i-hop fingerprint* of a vertex $v \in \mathrm{V}$, denoted $f_i(v)$, is the sequence

$$(\{S_u | u \in N(v,0)\}, \cdots, \{S_u | u \in N(v,i)\})$$

where $N(v,j)$ denotes the set of vertices whose minimum distance to v is j (the *jth-hop neighbours* of v.) We say an edge-labeled graph $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \Sigma)$ is *i-hop k-anonymous* if for each node $v \in \mathrm{V}$, there exist $k-1$ other nodes with the same i-hop fingerprint as v.

For this problem, as in the case of k-**LS-SAP**, we are given an edge labeled graph G and a subset of vertices X. We need to add the smallest number of edges to G to make X 1-hop k-anonymous. Similar to neighbourhood anonymity, we can reduce k-**Table Graph Anonymization** to 1-hop k-anonymity.

Lemma 7 *Given a table graph $\mathrm{G_T}$, X can be made k-label sequence anonymous by adding at most j edges if and only if it can be made 1-hop k-anonymous by adding at most j edges.*

Proof (If:) This direction of the proof is straightforward, because 1-hop anonymity implies label sequence anonymity.
(Only if:) By Lemma 5, k-label sequence anonymizing X optimally will result in the same set of adjacent vertices for every $y \in Y$, where Y is an equivalence class of X. Since the set of adjacent vertices is the same, the 1-hop fingerprint of every vertex $y \in Y$ is also the same. Therefore, 1-hop k-anonymity is NP-hard.

5.4 k-Symmetry Anonymization

k-Symmetry was introduced by Wu et al. [22]. Under this notion of anonymity, for each vertex v in the network, there exists at least $k-1$ other vertices which can act as an image of v under some automorphism of the modified network. To define the concept formally, we need the following definition:

Definition 16 (Automorphism Equivalence) Two vertices u, v of a graph $\mathrm{G} = (\mathrm{V}, \mathrm{E})$ are said to be *automorphically equivalent* if there is an automorphism of G that maps u to v. Automorphism equivalence is an equivalence relation on V and the partition of V induced by this equivalence relation is called the automorphism partition of G, denoted by Orb(G).

k-Symmetry anonymity requires that all orbits have size at least k. Formally we have:

Definition 17 (k-Symmetry Anonymity) A graph G is *k-symmetry anonymous* if $\forall \Delta \in \mathrm{Orb(G)}$, $|\Delta| \geq k$.

For this problem, as in the case of **k-LS-SAP**, we are given an edge labeled graph G and a subset of vertices X. We need to add the fewest number of edges to G to make X k-symmetry anonymous. Again, we can reduce **k-Table Graph Anonymization** to k-symmetry anonymity.

Lemma 8 *Given a table graph* $\mathrm{G_T}$*, X can be made k-label sequence anonymous by adding at most j edges if and only if it can be made k-symmetry anonymous by adding at most j edges.*

Proof (If:) It is easy to see that k-symmetry anonymity implies k-label sequence anonymity.
(Only if:) For k-symmetry anonymity, it is required that if $Y = \{y_1, y_2, \ldots, y_m\}$ is an equivalence class of X, then there is an automorphism of the anonymized graph that takes y_i to y_j for $1 \leq i, j \leq m$. This is the case for a table graph that is made k-label sequence anonymous in the optimal manner. Since, by Lemma 5, two vertices in Y are adjacent to the same set of neighbours, the mapping that maps y_i to y_j and vice versa and is the identity mapping on the rest of the vertices is an automorphism. Therefore, k-symmetry anonymity is NP-hard.

6 Bipartite Graphs

To recall, in the last section we demonstrated hardness for a series of k-anonymization problems on general edge-labeled graphs. In this section, we turn to the special case of bipartite graphs and establish that the hardness of bipartite graphs depends on the fixed value of k and whether the edges are labeled. This is an important result because bipartite graphs arise quite often in social networks that are based upon two distinct groups of entities (e.g., films and viewers, patients and drugs, teachers and students).

6.1 Edge-labeled Bipartite Graphs

We start with the edge-labeled setting by restating **k-LS-SAP** for bipartite graphs.

Problem 6 (k-Label-Sequence-Based Bipartite Subset Anonymization Problem (k-LS-BSAP)): *Given a labeled bipartite graph* $\mathrm{G} = ((\mathrm{V}, \mathrm{W}), \mathrm{E}, \Sigma)$ *and* $X \subseteq \mathrm{V}$, *find a bipartite graph* $\mathrm{G}' = ((\mathrm{V}, \mathrm{W}), \mathrm{E} \cup \mathrm{E}', \Sigma \cup \Sigma')$ *such that* X *is* k*-anonymous in* G' *and* $|\mathrm{E}'|$ *is minimized.*

*6.1.1 An Algorithm for k-**LS-BSAP** with $k = 2$*

We first show that the problem of finding an optimal 2-anonymization can be reduced to the problem of finding a min-cost perfect matching in a hypergraph containing edges of size 2 and 3. We then use a result shown by Anshelevich and Karagiozava [3] in a manner similar to Blocki and Williams [5] in order to conclude that there is a polynomial time algorithm for finding an optimal 2-anonymization. For simplicity, we will assume that $X = \mathrm{V}$. The algorithm we present below can be easily modified to work for any $X \subseteq \mathrm{V}$.

As stated by Liu and Terzi [16], we can assume that in any 2-anonymization of V every anonymous group is of size two or three (i.e., less than $2k = 4$). We construct a hypergraph $\mathrm{H} = (\mathrm{V}, \mathrm{E})$, where E contains every possible subset of V of size 2 and 3. We associate a cost $c(e)$ with each edge e in H. For any edge e, $c(e)$ will be the number of new edges that need to be added to make the vertices in e have the same label sequence so that they form an anonymous group. Let S_u denote the label sequence of a vertex u in V and for convenience of notation let us treat the label sequences as multisets and use multiset operations. Then,

$$\begin{aligned} c(\{u,v\}) &= |S_u \setminus S_v| + |S_v \setminus S_u|\,, \text{ if } |S_u \cup S_v| \leq |V|; \\ c(\{u,v\}) &= \infty, \text{ otherwise;} \end{aligned}$$

$$\begin{aligned} c(\{u,v,w\}) &= |(S_v \cup S_w) \setminus S_u| + |(S_u \cup S_w) \setminus S_v| \\ &\quad + |(S_u \cup S_v) \setminus S_w|\,, \text{ if } |S_u \cup S_v \cup S_w| \leq |V|; \\ c(\{u,v,w\}) &= \infty, \text{ otherwise.} \end{aligned}$$

In other words, the cost of creating an anonymous group of size two is the symmetric difference of the two label sequences. For example, if a label l occurs twice in a set S_u and once in another set S_v, then one of the two occurrences of the label l will be in $S_u \setminus S_v$. The cost of anonymizing three vertices u, v and w into one group is to add all the edges present in the union of two label sequences but not in the third. The cost is infinite, however, if there are too few vertices in V to anonymize u and v (and w) together.

Now, finding the optimal 2-anonymization reduces to finding a minimum-cost perfect matching in the edge weighted hypergraph H constructed above. A perfect matching in H is a set of edges such that every vertex in V is present in exactly one of the edges.

We recall the result of Anshelevich and Karagiozava [3]:

> *Given any hypergraph* H *with edges of size two and three with an associated cost function d on the edges of* H*, there is a polynomial time algorithm for finding a minimum-weight perfect matching in* H *provided d satisfies the following simplex condition: For any edge $e = \{u, v, w\}$ in* H *the edges $\{u, v\}, \{v, w\}, \{u, w\}$ are also in* H *and*
>
> $$d\{u, v\} + d\{v, w\} + d\{u, w\} \leq 2d\{u, v, w\}.$$

In order to use this result, we need the following, Lemma 9:

Lemma 9 *The cost function c satisfies the simplex condition of Anshelevich and Karagiozava [3]. That is,*

$$c\{u, v\} + c\{v, w\} + c\{u, w\} \leq 2c\{u, v, w\}.$$

Proof To show this, we will consider edge labels in three types of regions in the Venn diagram for the three (multi)sets S_u, S_v and S_w and show that their contribution to the LHS of the equation is at most their contribution to the RHS.

Regions of type 1 contain edge labels present in one of the three sets and not the other two. These labels contribute 2 to the LHS and 4 to the RHS. For example, suppose a label l is in S_u but not in S_v and S_w. Then, it contributes a cost of 1 to $c\{u, v\}$ and $c\{u, w\}$. On the other hand, it contributes 2 to $c\{u, v, w\}$ and hence 4 to $2c\{u, v, w\}$.

Regions of type 2 contain edges labels that are in two sets and not in the third. These labels contribute 2 to the LHS and 2 to the RHS. For example, suppose a label l is in S_u and S_v but not in S_w. Then, it contributes a cost of 1 to $c\{u, w\}$ and $c\{v, w\}$. On the other hand, it contributes 1 to $c\{u, v, w\}$ and hence 2 to $2c\{u, v, w\}$.

Regions of type 3 contain labels present in all the three sets. These labels do not contribute to either side.

Therefore, we have Theorem 2:

Theorem 2 *k-**LS-BSAP** is in* P *for $k = 2$.*

*6.1.2 A Hardness Result for k-**LS-BSAP***

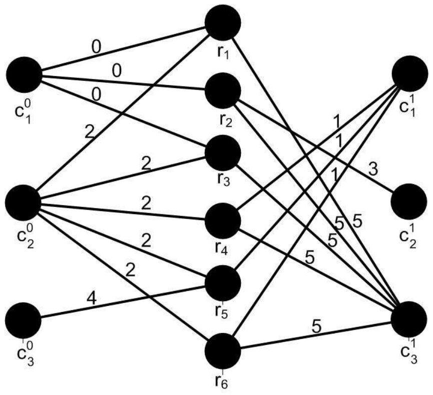
In the previous subsection we demonstrated that k-**LS-BSAP** is tractable when $k = 2$. In this subsection, on the other hand, we illustrate that for larger k, Theorem 3 holds:

Theorem 3 *k-**LS-BSAP** is NP-complete for $k \geq 3$.*

Proof We can build a reduction from the NP-hard table anonymization problem introduced in §3 to the decision version of k-**LS-BSAP**, and use similar techniques as in §5. We can build the reduction from k-**ENTRY-ANONYMITY** again and proceeds as follows:

Entry	A_1	A_2	A_3
1	0	0	1
2	0	1	1
3	0	0	1
4	1	0	1
5	1	0	0
6	1	0	1

(a) Example Table T



(b) T as a bipartite graph G_T

Fig. 4 Example of transforming a table T into a bipartite graph G_T.

Reduction: Our reduction is described as follows:

Given a Table T, let $T_{(m,j)} \in \{0, 1\}$ denote the value of attribute j in row m. Then, the edge-labeled bipartite graph G_T corresponding to T is constructed as follows:

- $V_T = \{r_1, r_2, \ldots, r_n\}$.
- $W_T = \{c_j^i | 1 \leq j \leq l, i \in \{0, 1\}\}$.
- Let $E_T = \{(r_m, c_j^i, 2(j-1)+i) | T_{(m,j)} = i\}$ where $1 \leq m \leq n$, $1 \leq j \leq l$ and $i \in \{0, 1\}$.
- $\Sigma_T = \{0, 1, \ldots, 2l-1\}$.
- $\mathcal{L}((r_m, c_j^i)) = 2(j-1)+i$ for $(r_m, c_j^i) \in E_T$.
- Finally, remove all isolated vertices from G_T.

In other words, we encode a binary table as a bipartite graph in which a row vertex $r_m \in V_T$ is connected to a column vertex $c_j^0 \in W_T$ (alt., $c_j^1 \in W_T$) with label $2(j-1)$ $(2(j-1)+1)$ if the (m, j)th entry of the table is 0 (1). See Figure 4.

Let $X = \{r_1, \ldots, r_n\}$ denote the set of row vertices of G_T. T can be k-anonymized by suppressing at most t entries if and only if we can k-anonymize X by adding at most t new labeled edges. The proof of this is immediate from Lemmata 1, 2, and 4 which we used in the proof of hardness of k-**LS-SAP**.

We do remark that the decision version of k-**LS-BSAP** is in NP. The certificate of membership is the same as in Theorem 1.

6.2 Unlabeled Bipartite Graphs

Here we explore the setting of unlabeled bipartite graphs, showing therein that degree-based subset anonymization is in P by constructing a polynomial-time algorithm. This problem has also been studied in the setting of vertex addition [8]. First, we restate ***k*-D-SAP** for the bipartite setting:

Problem 7 (*k*-Degree-Based Bipartite Subset Anonymization Problem(*k*-D-BSAP)): Given an unlabeled bipartite graph $\mathrm{G} = ((\mathrm{V}, \mathrm{W}), \mathrm{E})$ and $X \subseteq \mathrm{V}$, find a graph $\mathrm{G}' = ((\mathrm{V}, \mathrm{W}), \mathrm{E} \cup \mathrm{E}')$ such that, $\mathrm{E}' \subseteq \mathrm{V} \times \mathrm{W}$, X is k-anonymous in G' and the number of new edges added, $|\mathrm{E}'|$, is minimized.

We note that in this setting X is only allowed to be a subset of V. This is based on the understanding that for social networks represented as bipartite graphs, each side of the bipartition represents one type of entity and we are interested in anonymizing only one type. For example, in the Patient-Drug example, we are only interesting in anonymizing subsets of patients, not drugs.

*6.2.1 k-**D-BSAP** with $k = 2$*

We begin by first noting that the technique used to produce an algorithm for ***k*-LS-BSAP** with $k = 2$ in §6.1.1 can be simplified for ***k*-D-BSAP** to demonstrate that ***k*-D-BSAP** is in P for $k = 2$, as well. This is not surprising, since unlabeled graphs are a special case of labeled graphs where $\Sigma = \{\emptyset\}$. In this scenario, the new cost function is the difference in degrees. Recall that $d(u)$ denotes the degree of a vertex u.

$$\begin{aligned}
c'(\{u, v\}) &= |d(u) - d(v)|. \\
c'(\{u, v, w\}) &= [max(d(u), d(v), d(w)) - d(u)] \\
&\quad + [max(d(u), d(v), d(w)) - d(v)] \\
&\quad + [max(d(u), d(v), d(w)) - d(w)].
\end{aligned}$$

Lemma 10 *c' satisfies the simplex condition.*

Proof Without loss of generality, let $d(u) > d(v) > d(w)$. Then, the LHS evaluates to $d(u)-d(v)+d(v)-d(w)+d(u)-d(w) = 2(d(u)-d(w))$. Furthermore,

$$\begin{aligned}
RHS &= 2[(d(u) - d(u) + d(u) - d(v) + d(u) - d(w)] \\
&= 2[2d(u) - d(v) - d(w)] \\
&> 2[2d(u) - d(u) - d(w)] \\
&= 2[d(u) - d(w)]
\end{aligned}$$

Now, by using the result of Anshelevich and Karagiozava [3], we get that 2-anonymity for unlabeled bipartite graphs is in P.

*6.2.2 k-**D-BSAP** with $k \geq 2$*

Our main result in this section subsumes the previous case of §6.2.1. We show that there is an efficient algorithm for this problem using the techniques of Liu and Terzi [16], $\forall k \geq 2$. For simplicity, we will assume that $X = \mathrm{V}$. Our algorithm can be easily modified to work for any $X \subseteq \mathrm{V}$. Let $|\mathrm{V}| = n$ and let $d = (d_1, d_2, \ldots, d_n)$ be the degree sequence of the vertices of V. We start with the preprocessing step of sorting the degree sequence d in $\mathrm{O}(n \log n)$ time. The algorithm proceeds in two main steps:

- **Degree Anonymization:** Given d, the algorithm outputs the sequence $d' = (d'_1, d'_2, \ldots, d'_n)$ satisfying $d'_i \geq d_i$ such that $\Sigma_i(d'_i - d_i)$ is minimized and d' is k-anonymous.
- **Graph Construction:** The algorithm constructs the graph G' in which the degree of the vertex u_i is d'_i. In other words, V is k-anonymous in G'.

Degree Anonymization

We start with the following proposition which can be easily verified by noting that any anonymous group of more than $2k$ elements can be split into two disjoint anonymous groups, each of size at least k, such that the cost of the disjoint groups is less or equal to that of the singleton.

Proposition 1 *Without loss of generality, every anonymous group in d' is of size less than $2k$.*

Given a sorted degree sequence d, let $DA(d[1,i])$ denote the cost of k-anonymizing the subsequence $d[1,i]$. Also, let $C(d[i,j])$ be the cost of including the vertices $\{u_i, u_{i+1}, \ldots, u_j\}$ in one anonymous group. Clearly,

$$C(d[i,j]) = \Sigma_{l=i}^{j}(d(i) - d(j)).$$

Using the proposition above, we get the following dynamic programming equations to compute d'. In particular, for $i < 2k$:

$$DA(d[1,i]) = C(d[1,i]),$$

while for $i \geq 2k$,

$$DA(d[1,i]) = \min_{j \leq t \leq i-k}(DA(d[1,t]) + C(d[t+1,i])),$$

where $j = \max\{k, i - 2k + 1\}$.

The first equation uses the fact that if $i < 2k$, it is not possible to have more than one anonymous group. Therefore, the optimal cost of creating a single group involves making all the degrees equal to $d(1)$. The second equation says that if $i > 2k$, the degree anonymization cost consists of the degree anonymization cost of the subsequence $d[1,t]$ and the optimal cost of putting the vertices $t+1, \ldots, i$ into a single group. Moreover, this group has to be of size less than $2k$. The running time of this dynamic programming step is $\mathrm{O}(nk)$.

Graph Construction

We observe the following property of d'.

Lemma 11 *Let d_{max} be the maximum degree of a vertex in* V *in the graph* G. *Then $d'(i) < d_{max}$ for all i.*

Proof Suppose not for the sake of contradiction. Then, there is an anonymous group of d' in which all values are equal and greater than d_{max}. Replacing every entry by d_{max} will help produce a k-anonymous sequence with a lower degree anonymization cost.

Therefore, once we obtain the k-anonymous degree sequence d' from the previous step, the algorithm adds for each vertex $u_i \in \mathrm{V}$, $d'(u_i) - d(u_i)$ new edges from u_i to (arbitrary) vertices in W.

The running time of this step is $\mathrm{O}(nd_{max})$ where d_{max} is the maximum degree of a vertex in V.

In summary, we get the following theorem:

Theorem 4 ***k-D-BSAP*** $\in \mathrm{P}$, *for $k \geq 2$. In particular, there is an algorithm with running time $\mathrm{O}(n(k + d_{max}) + n \log n)$ that solves this problem.*

7 A Hardness Result for Attribute Disclosure

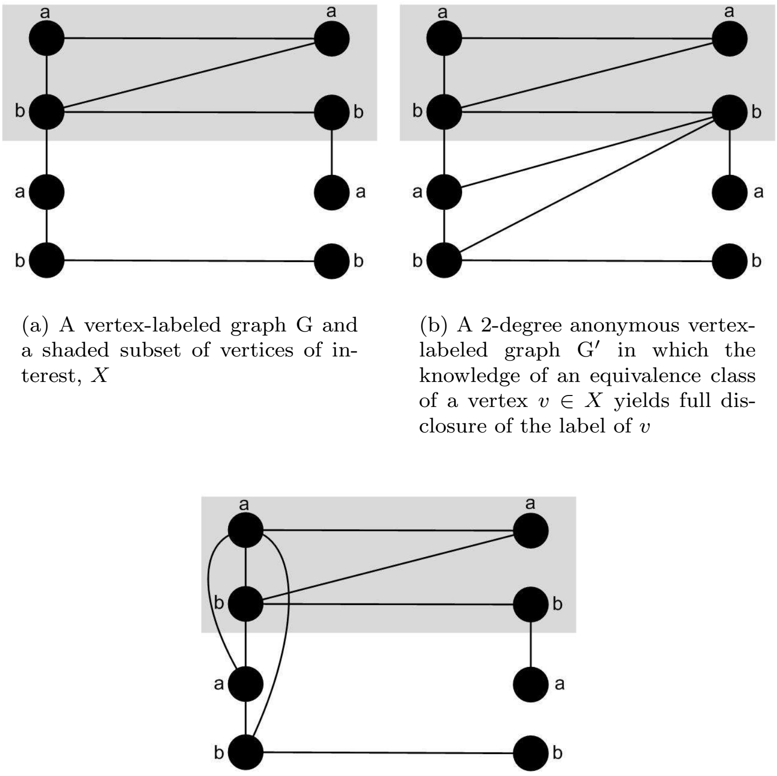
Throughout this paper, we have investigated labeled graphs, so it is natural that we should consider attribute disclosure attacks. While relatively novel in terms of graph anonymization, the problem is well studied within the table privacy community. We begin this section by motivating the need for protection against attribute disclosure attacks, prompting our definition of *t-closeness* for graphs, a natural analog of the measure by the same name within table privacy. We then give a hardness result for the problem.

7.1 Why t-closeness?

Example 4 Consider the example in Figure 5. The second graph (Figure 5(b)) is an example of an optimal 2-degree-anonymization of the shaded subset of the first graph (Figure 5(a)), but demonstrates the limitation of mere structural anonymization. Note that if the adversary knows the degree of a shaded target v in the anonymized graph he indeed cannot be certain which vertex corresponds to v with probability greater than .5, but he still absolutely knows the label of v: every shaded vertex in the anonymous graph with the same degree also has the same label.

To protect a sensitive label of a vertex, then, it is not enough to just conceal the identity of the vertex. It is also necessary to ensure that knowledge of an equivalence class (i.e., the assumed adversarial knowledge) is not sufficient to infer much new knowledge about the labels of vertices within that equivalence class. This is our motivation behind defining an attribute disclosure attack (Definition 11).

Consider now the third graph (Figure 5(c)) which is also an optimal 2-degree anonymization of the shaded subset of the first graph. In this case,



(a) A vertex-labeled graph G and a shaded subset of vertices of interest, X

(b) A 2-degree anonymous vertex-labeled graph G$'$ in which the knowledge of an equivalence class of a vertex $v \in X$ yields full disclosure of the label of v

(c) A 0-close, 2-degree anonymous vertex-labeled graph G$''$ for which even knowledge of the equivalence class of $v \in X$ offers no new information about the label of v

Fig. 5 An example of the susceptibility of k-anonymous graphs to attribute disclosure attacks and how t-closeness addresses that susceptibility

however, both equivalence classes have equal ‘a’ and ‘b’ labels, just as in the overall graph. Thus, if the adversary can identify the equivalence class of his target vertex, he still cannot infer new knowledge about the sensitive label. In this case, the anonymization procedure has achieved 0-closeness (Definition 12) because the distance between distributions of labels in the original graph and each equivalence class is upper bounded by 0.

7.2 A Hardness Result for t-closeness

Here, we present a hardness result for t-closeness conjoined with k-**LS-SAP**. We start by describing the problem we study precisely: Given a vertex-labeled graph $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \Sigma, \ell)$ and parameters m and t, is it possible to convert G into a graph $\mathrm{G}' = (\mathrm{V}, \mathrm{E} \cup \mathrm{E}', \Sigma, \ell)$ such that G' is t-close and $|\mathrm{E}'| \leq m$? We show here that this problem is NP-complete.

To do so, we begin by defining a new anonymization problem, called ***k*-VLS-AP**:

Problem 8 (*k*-Vertex-Labeled Sequence Anonymization Problem (*k*-VLS-AP)):Given a vertex-labeled graph $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \Sigma, \ell)$, can G be converted to a graph $\mathrm{G}' = (\mathrm{V}, \mathrm{E} \cup \mathrm{E}', \Sigma, \ell)$ that is k-label-sequence anonymous such that $|\mathrm{E}'| \leq m$?

Lemma 12 *k-**VLS-AP** is NP-complete for $k \geq 3$.*

Proof This result follows by looking into the proof of Theorem 1 of Zhou and Pei [25]. Our main observation is that though Theorem 1 [25] shows a hardness result for the notion of neighbourhood anonymity in vertex-labeled graphs, it is easily seen from the proof details that, in fact, it shows the hardness of label sequence anonymity in vertex-labeled graphs under edge additions.

NP-hardness for ***k*-VLS-AP** helps us show the hardness of t-closeness using ideas similar to Theorem 2 from the same work of Zhou and Pei [25].

Theorem 5 *t-Closeness is NP-complete if the equivalence classes are required to be k-vertex label sequence anonymous for $k \geq 3$.*

Proof We reduce the ***k*-VLS-AP** problem to t-closeness in social networks. Suppose that for a given vertex labeled graph G, we want to check if it can be made k label sequence anonymous by adding at most m edges. We construct a new vertex-labeled G' by assigning to each vertex in G a new unique sensitive label. That is, every vertex will now have a tuple (l_1, l_2) as its label. Here, l_1 is the old label present in G and l_2 is the new sensitive label we have introduced. Now we can check that G can be converted into a k label sequence anonymous graph G_1 such that $|\mathrm{E}(\mathrm{G}_1) - \mathrm{E}(\mathrm{G})| \leq m$ if and only if G' can be converted into a t-close graph G_2 such that $|\mathrm{E}(\mathrm{G}_2) - \mathrm{E}(\mathrm{G}')| \leq m$ and $t = 2\left(1 - \frac{k}{n}\right)$. Here, t is the distance between the two probability distributions $(\frac{1}{n}, \ldots, \frac{1}{n})$ and $(\frac{1}{k}, \ldots, \frac{1}{k}, 0, \ldots)$.

The other direction of the proof follows trivially from the fact that t-closeness with $t = 2\left(1 - \frac{k}{n}\right)$ implies vertex label sequence anonymization because the latter is a prerequisite condition of the former.

8 Conclusion

Data privacy is of paramount importance, especially while social networks indefinitely grow in user base, data collection, and analysis opportunity. In this paper, we have initiated a systematic study of the hardness of providing this privacy when graph data is to be released to third parties. We have made a series of contributions: establishing a framework for proving hardness of k-**SAP** problems, defining t-closeness to better protect against attribute disclosure attacks on vertex-labeled graphs, and ascertaining the following complexity results:

- For general, edge-labeled graphs, label sequence subset anonymization–and thus table graph anonymization, k-neighbourhood anonymity, i-hop anonymity, and k-symmetry–are NP-complete for $k \geq 3$;
- For bipartite, edge-labeled graphs, label sequence subset anonymization is in P for $k = 2$ and is NP-complete for $k \geq 3$;
- For bipartite, unlabeled graphs, degree-based subset anonymization is in P for all values of k;
- And for general, vertex-labeled graphs, we show that vertex label sequence-based anonymization and consequently t-closeness is NP-complete.

In addition to these results, we have also implicitly contributed the identification of several open problems related to graph anonymization. In particular, it is still unknown whether:

- There is a polynomial time algorithm for label sequence subset anonymization when $k = 2$;
- Effective approximation algorithms exist for any of these hard problems; or
- t-Closeness is NP-complete when conjoined with preventitive measures of identity disclosure other than vertex label sequence anonymization.

References

1. Abdallah, S.: Generalizing unweighted network measures to capture the focus in interactions. Social Network Analysis and Mining **1**(4), 255–269 (2011)
2. Aggarwal, G., Feder, T., Kenthapadi, K., Motwani, R., Panigrahy, R., Thomas, D., Zhu, A.: Anonymizing tables. In: Proc. International Conference on Database Theory (ICDT), pp. 246–258 (2005)
3. Anshelevich, E., Karagiozova, A.: Terminal backup, 3d matching, and covering cubic graphs. In: Proc. ACM Symposium on Theory of Computing (STOC), pp. 391–400 (2007)
4. Backstrom, L., Dwork, C., Kleinberg, J.M.: Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In: Proc. Conference on World Wide Web (WWW), pp. 181–190 (2007)
5. Blocki, J., Williams, R.: Resolving the complexity of some data privacy problems. In: Proc. International Colloquium on Automata, Languages and Programming, pp. 393–404 (2010)
6. Bonizzoni, P., Vedova, G.D., Dondi, R.: The k-anonymity problem is hard. In: Fundamentals of Computation Theory (FCT), pp. 26–37 (2009)
7. Cha, M., Pérez, J., Haddadi, H.: The spread of media content through blogs. Social Network Analysis and Mining pp. 1–16. URL `http://dx.doi.org/10.1007/s13278-011-0040-x`. Preprint
8. Chester, S., Kapron, B., Ramesh, G., Srivastava, G., Thomo, A., Venkatesh, S.: k-anonymization of social networks by vertex addition. In: Advances in Databases and Information Systems, ADBIS (2011)

9. Chester, S., Srivastava, G.: Social network privacy for attribute disclosure attacks. In: Proc. Advances in Social Networks Analysis and Mining (ASONAM) (2011)
10. Cormode, G., Srivastava, D., Yu, T., Zhang, Q.: Anonymizing bipartite graph data using safe groupings. Very Large Databases Journal (VLDBJ) **19**(1), 115–139 (2010)
11. Fung, B.C.M., Wang, K., Fu, A.W.C., Pei, J.: Anonymity for continuous data publishing. In: Proc. International Conference on Extending Database Technology (EDBT), pp. 264–275 (2008)
12. Gionis, A., Tassa, T.: k-anonymization with minimal loss of information. In: Proc. European Symposium on Algorithms (ESA), pp. 439–450 (2007)
13. Hay, M., Miklau, G., Jensen, D., Towsley, D.F., Weis, P.: Resisting structural re-identification in anonymized social networks. Proc. Very Large Databases (PVLDB) **1**(1), 102–114 (2008)
14. Kapron, B., Srivastava, G., Venkatesh, S.: Social network anonymization via edge addition. In: Proc. Advances in Social Networks Analysis and Mining (ASONAM) (2011)
15. Li, N., Li, T., Venkatasubramanian, S.: t-closeness: Privacy beyond k-anonymity and l-diversity. In: Proc. International Conference on Data Engineering (ICDE), pp. 106–115 (2007)
16. Liu, K., Terzi, E.: Towards identity anonymization on graphs. In: Proc. ACM Special Interest Group on Management of Data (SIGMOD), pp. 93–106 (2008)
17. Machanavajjhala, A., Kifer, D., Gehrke, J., Venkitasubramaniam, M.: l-diversity: Privacy beyond k-anonymity. ACM Transactions on Knowledge Discovery from Data (TKDD) **1**(1) (2007)
18. Meyerson, A., Williams, R.: General k-anonymization is hard. In: Proc. Principles of Database Systems (PODS) (2004)
19. Sweeney, L.: Achieving k-anonymity privacy protection using generalization and suppression. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **10**(5), 571–588 (2002)
20. Thompson, B., Yao, D.: The union-split algorithm and cluster-based anonymization of social networks. In: Proc. ACM Symposium on Information, Computer and Communications Security (ASIACCS), pp. 218–227 (2009)
21. Tripathy, B.K., Panda, G.K.: A new approach to manage security against neighborhood attacks in social networks. In: Proc. Advances in Social Networks Analysis and Mining (ASONAM), pp. 264–269 (2010)
22. Wu, W., Xiao, Y., Wang, W., He, Z., Wang, Z.: k-symmetry model for identity anonymization in social networks. In: Proc. International Conference on Extending Database Technology (EDBT), pp. 111–122 (2010)
23. Yuan, M., Chen, L., Yu, P.S.: Personalized privacy protection in social networks. Proc. Very Large Databases (PVLDB) **4**(2), 141–150 (2010)
24. Zheleva, E., Getoor, L.: Preserving the privacy of sensitive relationships in graph data. In: Proc. Privacy, Security, and Trust in KDD (PinKDD), pp. 153–171 (2007)
25. Zhou, B., Pei, J.: The k -anonymity and l -diversity approaches for privacy preservation in social networks against neighborhood attacks. Knowledge and Information Systems **28**(1), 47–77 (2011)
26. Zweig, K., Kaufmann, M.: A systematic approach to the one-mode projection of bipartite graphs. Social Network Analysis and Mining **1**(3), 187–218 (2011)