Uncertain Graphs Sharing Using Synthetic Private Graph Models

Dongqing Xiao, Mohamed Y. Eltabakh, Xiangnan Kong

Computer Science Department, Worcester Polytechnic Institute
Worcester, United States of America
{dxiao, meltabakh, xkong} @wpi.edu

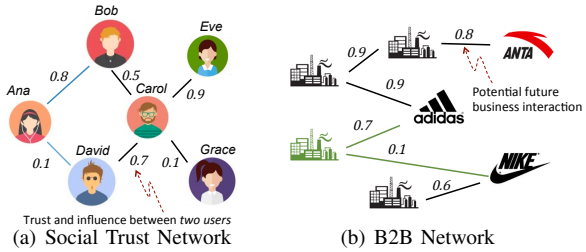
***Abstract*—Research on social and business applications requires open access to real datasets. Such datasets can be shared, generally in the form of uncertain graphs whose edges are labeled with a probability of existence. While releasing uncertain graphs often risks exposing sensitive user information to the public. Current works are based on deterministic graphs that overlook the inherent uncertainty in edges. To overcome such limitation, our work seeks a solution to release *uncertain graphs* with high utility while preserving privacy. We show that simply combining conventional graph anonymization with representative extraction strategy results in the addition of noise that significantly disrupt *uncertain graph* structure, degrading its utility. Instead, we introduce an uncertainty-aware XXXX**

I. Introduction

In many prevalent application domains, such as business to business (B2B) [25], social networks [2], [21], and sensor networks [39], graphs serve as powerful models to capture the complex relationships inherent in these applications. Most graphs in these applications are uncertain by nature, where each edge carries a degree of uncertainty (probability) representing the probability of its presence in the real world. This uncertainty can be due to various reasons ranging from the use of prediction models to predict the edges (as in social media and B2B networks) to physical properties that affect the edges' reliabilities (as in sensor and communication networks).

These uncertain graphs are of significant importance to support various data mining tasks *e.g.*, understanding graph structures [8], [22], social interactions [11], information discovery and propagation [40], advertising and marketing [21], among many others. When compared to sharing the results of data mining, data publishing gives greater flexibility because recipients can perform unlimited analysis, data explosion and novel methods. However, the publishing of these uncertain graphs might violate participants' privacy due to the existence of sensitive information.

Motivation Scenario I (Social Trust Networks): *In social networks, the trust and influence relationships among users—which may greatly impact users' behaviors—are usually probabilistic and uncertain [21] (See Figure 1(a)). The existence of the trust relationship depends on many factors, such as the area of expertise and emotional connections. Researchers are very interested in studying the structure of social trust networks, in order to promote products, or choose strategies for a campaign. However, the release of such uncertain graphs with simple anonymization may cause serious privacy issues.*



(a) Social Trust Network (b) B2B Network

Figure 1: Examples of real-world uncertain graphs with privacy concerns.

The attackers can re-identify private and sensitive information, such as the identity of the users and their trustiness relationship, from the released data.

Motivation Scenario II (B2B Networks): *Another uncertain graph example comes from Businesses to Businesses network (See Figure 1(b)). In these networks, e.g., "Alibaba", nodes represent companies (or businesses in general) while edges represent the trust and the potential of future transactions among them [25]. Such future interactions are uncertain since they are obtained by prediction models based on historical data [24]. B2B networks can be analyzed and mined for various applications including advertisement targeting [1] and customer segmenting [3]. Certainly, information about a company's interactions with other companies is considered sensitive data since any leak can be used to infer the company's financial conditions.*

These scenarios show the immediate need for efficient methods privacy preserving uncertain graph publishing, where sanitation or anonymization is applied to the input uncertain graph before publishing. Despite the number of graph anonymization techniques have been proposed [7], [9], [26]–[30], [38], the ignorance of edge uncertainty makes them inefficient for uncertain graph sanitation task. The inefficiency can be due to various reasons ranging from wrong assumption of privacy attacks to improper utility loss metrics. More specifically, the key new challenges in the context of uncertain graphs include:

However, in the uncertain case, the revealed edge uncertainties make the two nodes follow different degree distributions, and thus the adversary has more confidence (around 90%) that `Ana` maps to Node `a`.

This example shows that the release of the associated edge uncertainty increases the potential privacy risk. Therefore, uncertain graph anonymization must take into consideration these edge uncertainties, otherwise, it will fail to protect the privacy correctly. Evidently, ignoring the probabilities altogether and not adding them to the released graph in Figure **??**(b) is not a practical solution as it severely destroys the structure and the utility of the original graph.

• **Appropriate Utility Loss Metric for Uncertain Graphs:** Ideally, the anonymized graph should preserve the privacy with the smallest utility loss for permitting meaningful analysis tasks. Thus, it is crucial to understand and model the utility loss through well-defined metrics. The utility loss metric acts as a safeguard in anonymization process. Various metrics have been proposed for deterministic graphs such as the total number of edge modification [7], [26], spectrum discrepancy [38], community reconstruction error [30], [35], and shortest path length discrepancy [27]. These utility loss metrics have a clear and precise definition in the context of deterministic graphs, which is not the case for uncertain graphs. It is important to understand and model the key properties of *uncertain graph* to be preserved for analysis tasks. Then, the corresponding derived utility loss metric should be incorporated into uncertain graph anonymization as replacement of the aforementioned classical ones.

• **Increased Exponential Complexity of Uncertain Graph Anonymization:** The problem of k-anonymizing a given deterministic graph by as few graph contractions (edge addition, edge deletion, vertex addition and vertex deletion) as possible is shown to be NP-hard [16]. Existing techniques usually rely on heuristics to avoid combinatorial intractability. In uncertain graphs, the problem is even harder since an edge operation is no longer a binary operation (addition or deletion), but there can be infinite probability values that can be assigned to each edge. Therefore, efficient *uncertainty-aware* heuristics need to be developed to bring the solution to the realm of feasibility.

In this paper, we present the *"Chameleon"* framework for addressing the aforementioned challenges. Chameleon incorporates edge uncertainties into the core of the anonymization processing such as evaluations of privacy gain and utility loss. In contrast to the classical deterministic graph utility metrics, we propose a new utility metric based on the *reliability* measure—which is a core metric in numerous uncertain graph applications [4], [15], [40]. The anonymization process need to change the graph structure by modifying the edge probabilities of a subset of the edges, which is an exponential search space. Therefore, we propose a ranking algorithm that ranks the edges w.r.t the impact of a change on the graph structure—which we refer to as *"reliability Relevance"*—and that ranking will guide the edge selection process. Moreover, we propose a theoretically-founded probability-alteration strategy based on the entropy of graph degree sequence, which enables achieving maximum privacy gain for an added amount of perturbation.

In summary, the key contributions of this paper are the following:

- Identifying the new and important problem of uncertain graph anonymization where edge uncertainties need to be seamless integrated into the core of the anonymization process. Otherwise, either the privacy will not be protected or the utility will be severely damaged.
- Proposing a new utility-loss metric based on the solid connectivity-based graph model under the possible world semantics, namely the *reliability discrepancy* (Section III).
- Introducing a theoretically-founded criterion, called *reliability relevance*, that encodes the sensitivity of the graph edges and vertices to the possible injected perturbation. The criterion will guide the edges' selection during the anonymization process (Section **??**).
- Proposing uncertainty-aware heuristics for efficient edge selection and noise injection over the input uncertain graph to achieve anonymization at a slight cost of reliability (Section **??**).
- Building the Chameleon framework that integrates the aforementioned contributions. Chameleon is experimentally evaluated using several real-world datasets to evaluate its effectiveness and efficiency. The results demonstrate a significant advantage over the conventional methods that do not directly consider edge uncertainties (Section **??**).

II. Related Works

A significant amount of prior work has been done protecting privacy of network datasets. We summarize them here and clarify our privacy goals in this paper.

Synthetic Privacy. Early works on privacy-preserving network publishing mainly focus on developing anonymization techniques for deterministic graphs. Their goal is to *publish* the data in an anonymized manner without making any assumptions of the type of analysis and queries that will be executed on it. Once the data is published, it is available for any type of analysis. Most of them leverage *synthetic* privacy models derived from k-anonymity to create k identical neighborhoods, or k identical degree nodes.

Graph Anonymization Techniques. Current methods for anonymizing "graphs" can be classified into four main categories: (1) Clustering-based generation [6], [17], [18]; (2) *Edge modification* [26], [33], [35], [36], [41], (3) *Edge randomization* [27], [30], [38], and (4) *Uncertainty semantic-based modifications* which add uncertainty to some edges and thus converting the graph to an uncertain version [7], [29]. The uncertainty semantic-based approaches transform the original deterministic graph into an uncertain one to be published. These techniques are known as the state-of-art for their excellent privacy-utility tradeoff brought by the fine-grained perturbation leveraging the uncertain semantics. As ever mentioned, these techniques are tailored to *deterministic* graphs (unweighted & weighted) that overlook edge uncertainty.

Diffiential Privacy. The recent research on applying differential privacy to network data roughly falls into two directions. The first direction aims to release certain differentially private *data mining results*, such as degree distributions, sub-graph counts and frequent graph patterns. Such methods that release only query results require tracking the results: early uses of the data can affect the quality of later uses, thus no new queries can be permitted on the data. The second direction aims to publish a sanitized graph. Most research in this direction projects an input graph to dK-series and ensures differential privacy on dK-series statistics. These private statistics are then either fed into generators or MCMC process to generate a fit synthetic graphs. While current techniques are still inadequate to provide desirable data utility for many graph mining tasks.

Our Goal: Data Model & Privacy Policy In this work, we study the problem of releasing *uncertain* graph without violating privacy regulation. Conceptually, the problem can be interpreted as a natural generalization of the *determinitic* graph contexts to a larger probabilistic context, with the anonymization process being specifically optimized. Here, we remind the reader the approach that first casting the probability of every edge into a weight then apply existing anonymization strategy on this weighted graph to attain anonymized uncertain graph is problematic. First, there is no meaningful way to perform such casting. The casting has been proven to be erroneous in various uncertain graph mining tasks [32], [40]. What's more, there is no principled way to additionally encode normal weights on the edge. For example, each link in the road network can be weighted indicating the distance or travel time between them, and a probability can be assigned to model the likelihood of a traffic jam [20]. Thus, existing strategies for weighted graphs anonymization cannot be applied to *uncertain*graphs.

In the context of privacy preserving graph publishing, we can choose to adopt the synthetic or differential privacy policy. ϵ-differential privacy does relate to individual identifiability and provides strong privacy guarantee without making any assumption of privacy risks. While, there is no clear way to set a general policy for a value ϵ that provides sufficient privacy [23]. In contrast to ϵ-differential privacy, synthetic privacy model can generally be defined and understood based on the data schema; parameters have a clear privacy meaning that can be understood independent of the actual data and have a clear relationship to the legal concept of individual identifiability of data. In this work, we choose k-obfuscation, a variant of k-anonymity as our privacy model for uncertain graph *publishing*.

III. Problem Definition

In this section, we present the notations, definitions and the problem formulation.

A. Uncertain Graph

An uncertain graph $\mathcal{G} = (V, E, p)$, is defined over a set of nodes V, a set of edges E, and a set of probabilities p of edge existence. Following the literature, we consider the edge probabilities independent [19], [20], [32], [40], and we assume *possible-worlds* semantics [12]. Specifically, the *possible world* semantics interprets $\mathcal{G}$ as a set of possible deterministic graphs $W(\mathcal{G}) = \{G_1, G_2, ..., G_n\}$, where each deterministic graph $G_i \in W(\mathcal{G})$ includes all vertices of $\mathcal{G}$ and a subset of edges $E_{G_i} \subset E$. The probability of observing any possible world $G_i = (V, E_{G_i}) \in W(\mathcal{G})$ is

$$Pr[G_i] = \prod_{e \in E_{G_i}} p(e) \prod_{e \in E \setminus E_{G_i}} (1 - p(e))$$

In this work, we assume the input uncertain graph undirected and contains no self-loops or multiple edges.

B. Reliability-Based Utility Loss Metric

A well-chosen utility-loss metric may lead to substantially less sanitized graphs at a minimal loss of information. As be known to all, connectivity is a fundamental graph property and plays an important role in graph mining tasks such as locating $k-$nearest neighbor [32], graph clustering [4] and shortest paths detecting [40]. The connectivity model has been shown to be able to yield better representation than degree sequence model. The connectivity discrepancy was proven to be a proper utility-loss metric. In this paper, we use its generalized version – Reliability Discrepancy as the utility-loss metric in the uncertain graph context.

In uncertain graphs, the concept of reliability is used to generalize *connectivity* by capturing the probability that two given (sets of) nodes are reachable over all possible worlds of the uncertain graph as follows:

Definition 1. ***Two-Terminal Reliability [12]*** *Given an uncertain graph $\mathcal{G}$, and two distinct nodes u and $v \in V$, the reliability of (u, v) is defined as:*

$$R_{u,v}(\mathcal{G}) = \sum_{G \in W(\mathcal{G})} \mathcal{I}_G(u, v) Pr[G]$$

where $\mathcal{I}_G(u, v)$ is 1 iff u and v are contained in a connected component in G, and 0 otherwise.

Definition 2. ***Graph Reliability Discrepancy*** *The reliability discrepancy of graph $\tilde{\mathcal{G}} = (V, E, \tilde{p})$, denoted as $\Delta(\tilde{\mathcal{G}})$, w.r.t. an original graph $\mathcal{G} = (V, E, p)$ is defined as the sum of the two-terminal reliability discrepancy over all node pair $(u, v) \in V_{\mathcal{G}}$.*

$$\Delta(\tilde{\mathcal{G}}) = \sum_{(u,v) \in V_{\mathcal{G}}} |R_{u,v}(\mathcal{G}) - R_{u,v}(\tilde{\mathcal{G}})|$$

C. Attack Model and Privacy Criteria

In this paper, we focus on the "identity disclosure problem" [26] over uncertain graphs, which is one serious privacy leak concern when a graph dataset is published. Formally, give a published graph G, if and adversary can locate the target entity t as a vertex v of G with a high probability via auxiliary information, we said that the identity of t is disclosed. The popular assumption of auxiliary information is node degree [26].

Following the literature, we adopt the syntactic (k, ϵ)-*obfuscation* criterion [7] for privacy guarantee. Analogous to the well known k-anonymity notion, k-obf

requires to blend every vertex with *other* fuzzy-matching nodes. Compared to k-anonymity, k-obf, which is global and entropy-based quantification, is more adequate than the previous used local quantification based on a posteriori belief probabilities. An excellent discussion on k-obf was presented by Bonchi *et al.* [9]. Moreover, the introduction of a tolerance parameter ϵ, which allows skipping up to $\epsilon * |V|$ nodes, makes it more practical. The skipped nodes might be extreme unique nodes, e.g., Trump in a Twitter network, whose obfuscation is almost impossible. The formal definition is as follows:

Definition 3. ***(k, ϵ)-obf [7]*** *Let P be a vertex property (i.e., vertex degree in our work), $k \geq 1$ be a desired level of anonymity, and $\epsilon > 0$ be a tolerance parameter. An sanitized uncertain graph $\tilde{\mathcal{G}}$ is said to k-obfuscate a given vertex $v \in \mathcal{G}$ w.r.t P if the entropy $H()$ of the distribution $Y_{P(v)}$ over the nodes of $\tilde{\mathcal{G}}$ is greater than or equals to $\log_2 k$:*

$$H(Y_{P(v)}) \geq \log_2 k.$$

The uncertain graph $\tilde{\mathcal{G}}$ is (k, ϵ)-obf w.r.t property P if it k-obfuscates at least $(1 - \epsilon)|V|$ nodes in $\mathcal{G}$.

D. Problem Statement

Given the above foundation, we can now formulate the addressed problem.

Problem 1. ***Reliability-Preserving Uncertain Graph Anonymization*** *Given an uncertain graph $\mathcal{G} = (V, E, p)$ and anonymization parameters k and ϵ, the objective is to find a (k, ϵ)-obfuscated uncertain graph $\tilde{\mathcal{G}} = (V, E, \tilde{p})$ with minimal $\Delta(\tilde{\mathcal{G}})$. That is:*

$$\begin{aligned} &\underset{\tilde{\mathcal{G}}}{\operatorname{argmin}} && \Delta(\tilde{\mathcal{G}}) \\ &\textit{Subject to} && \tilde{\mathcal{G}} \textit{ is } (k, \epsilon) - obf \end{aligned}$$

IV. The Representative Anonymization Algorithm

Instead of designing new methods for *uncertain* graphs, we first consider somehow utilizes methods for deterministic graphs. Fortunately, there has been extensive work on extracting a single representative instance (deterministic one) of uncertain graphs that capturing graph statistics such as the expected vertex degrees [31].

Motivated by the preceding, in this work we introduce the representative anonymization (Rep-An) algorithm that combines isolated but complementary work from literature for uncertain graph anonymization. As shown in Figure2, we first extract a single *representative* instance from an original uncertain graph. Then, conventional anonymization techniques can be then applied on this representative to attain closely approximate anonymized output of the original uncertain graph. This body of research comes to its aid that anonymization can be carried out on uncertain graphs, regardless of the uncertainty inherent in the data.

However, this approach has several limitations. First, the input edge uncertainties (probabilities) are no longer integrated into the anonymization process since they are detached from the graph in the first step. Second, the anonymization process (the second step) is oblivious to the *reliability* metric since its input is a made-up deterministic graph. Third, since the two phases are isolated from each other, different phases are optimized for different metrics. As the result, this naive `Rep-An` approach introduces a high level of noise and consequently deteriorates the overall utility of the anonymized graph. In the experiment section, we further study this approach empirically and confirm its impracticality.

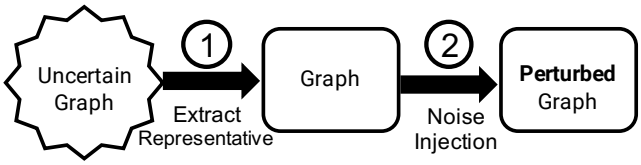


Figure 2: Overview of Rep-An. Noise is added to the extracted *representative* instance.

Table I: Characteristics of the datasets and privacy parameters

Graph	Nodes	Edges	Edge Prob	Tolerance level
DBLP	824,774	5,566,096	0.46	10^{-4}
Brightkite	58,228	214,078	0.29	10^{-3}
PPI	12,420	397,309	0.29	10^{-2}

A. Validation on Real Uncertain Graphs

In this section, we empirically evaluate the impact of noise injected to extracted *representative* instances by executing Rep-An on real uncertain graphs.

Methodology. We use three uncertain graphs that capture different real-world scenarios and have been used in prior uncertain graph mining studies. Table I lists uncertain graphs and their tolerance parameters used in our evaluation.

DBLP is a co-authorship network where the probability of an edge between two authors represents the likelihood two authors will collaborate in the future. The probability is obtained by a predictive model based on historical co-authorship data. [20].

BRIGHTKITE is a location-based social network where the probability of an edge between two users corresponds to the chance that two users visit each other. The probability can be obtained by a prediction model based on historical data [11].

PPI is a dataset of protein-protein interactions, provided by Disease Module Identification DREAM Challenge, where the probability of any edge corresponds to the confidence that the interaction actually exists. The probability is obtained through biological experiments.

We extract the single representative instances for each uncertain graph, introduce noise using the state-of-art graph anonymization strategy, and then compute the Reliability Discrepancy between the perturbed graph and the original uncertain graph as a measure of the level of graph structural error introduced. We approximate its expectation value by the average value obtained over the sampled possible worlds. Here, we use 1,000 samples since it has been shown that 1000 usually suffices to achieve accuracy converge [32].

Results. Figure **??** shows that the Rep-An algorithm produces a large error for large values of k (i.e. strong privacy

guarntees). As we mentioned, the low level of utility is due to the large noise Rep-An injects into the representative instance, resulting in a perturbed graph (uncertain one) which is significantly different from the original *uncertain* graph.

The high discrepancy is largely due to the representative extraction step. In the extreme case where $k = 1$ (i.e. no privacy guarntees), the sole representative extraction step produces high reliability errors. To better understand its limitation, we report the potential lower-bound on the error that could be achieved via Chamelon methods in Figure **??**. Indeed, Figure **??** shows that the utility loss of the Rep-An strategy can largely be attributed to the overlook of edge uncertainty.

V. Privacy Via Chameleon

The results in the previous section demonstrate the huge utility loss in the perturbed representative instance after adding noise to provide privacy guarantee. In this section, we propose a novel uncertainty-aware algorithm called Chameleon that enables uncertainty-aware control over the noise injected into the *original uncertain* graph. This qualifies Chameleon to provide enough privacy guarantee in better utility.

A. The Chameleon framework

Algorithm 1 Chameleon Iterative Skeleton

Input: Uncertain graph $\mathcal{G}$, adversary knowledge $\mathcal{K}$, obfuscation level k, tolerance level ϵ, size multiplier c and white noise level q
Output: The anonymized result $\tilde{\mathcal{G}}_{obf}$

$\sigma_l \leftarrow 0$; $\sigma_u \leftarrow 1$
repeat
  $\langle \tilde{\epsilon}, \tilde{\mathcal{G}} \rangle \leftarrow \texttt{GenObf}(\mathcal{G}, k, \epsilon, c, q, \sigma_u, \mathcal{K})$
  if $\tilde{\epsilon} = 1$ (fail) **then** $\sigma_l \leftarrow \sigma_u$; $\sigma_u \leftarrow 2\sigma_u$
until $\tilde{\epsilon} \neq 1$
repeat
  $\sigma \leftarrow (\sigma_u + \sigma_l)/2$
  $\langle \tilde{\epsilon}, \tilde{\mathcal{G}} \rangle \leftarrow \texttt{GenObf}(\mathcal{G}, k, \epsilon, c, q, \sigma_u, \mathcal{K})$
  if $\tilde{\epsilon} = 1$ **then** $\sigma_l \leftarrow \sigma$
  else $\sigma_u \leftarrow \sigma$; $\tilde{\mathcal{G}}_{obf} \leftarrow \tilde{\mathcal{G}}$
until $\sigma_u - \sigma_l$ is enough small
return $\tilde{\mathcal{G}}_{obf}$

We now introduce the state-of-art perturbation algorithm [7] that computes the noise needed to injected into the input *deterministic* graph to obtain the desired privacy level. Each selected edge is altered based on a stochastic variable drawn from a trunated Normal distribution, $R(\sigma)$. This distribution has density function proportional to the Normal distribution, with mean 0 and variance σ^2. Thus, small values of σ contribute towards better utility, but at the same time they provide lower level of obfuscation. Targeting for high utility, the algorithm aims at injecting the minmal amount of noise need to achieve the required obfuscation. Its computation is achieved via a binary search on the value of the noise uncertainty parameter σ, as shown in Algorithm **??**.

The core function of this process is the `GenObf` function which performs following core steps:

- Select a subset of edges subjects to further alteration;
- Alter selected edges as the computed amount of noise;

Contribution The conventional schemes are plausible if the operating edge probability is binary, which is unreasonable when dealing with uncertain graphs. Our goal is to develop a privacy mechanism that reduced the amount of noise that must be added to achieve a given privacy level for *uncertain* graphs. Our insight is to shift an existing framework for anonymizing *probabilistic* graphs by integrating uncertainty semantics into two core steps.

B. Uncertainty-aware Edge Selection

Figuring out the optimal subset of edges that balances the privacy gain and the utility loss is a typical combinational optimization problem. It involves the consideration over the exponential number of edge combinations. Let alone the infinite possibilities of probability values on the selected edges, which further complicates the problem.

To alleviate combinational intractability, the heuristics that have been proposed so far for anonymizing deterministic graphs can be classified into two main categories: (1) *Anonymity-oriented* heuristics that suggest injecting larger perturbations to the edges associated with the less-anonymized (more-unique) nodes [7], [10], [13], [26], [27], [30], [34]–[38], [41], and (2) *Utility-oriented* heuristics that suggest avoiding perturbations over *"bridge"* and sensitive edges whose deletion or addition would significantly impact the graph structure [10], [13], [27], [30], [35], [36], [38]. Note that these two types are complementary to each other and combining them would introduce an added benefit as confirmed by practice in deterministic graph anonymization [10]. Nevertheless, these two types of heuristics and their combination have not been explored yet in the context of *uncertain* graphs.

In this paper, we first extend the idea of *uniqueness* score via density estimation. Second, we propose a novel edge relevance that extends well-known graph concepts, such as "cut edge" for estimating structural errors incurred by edge probability alterations. In order to compute edge relevance in *uncertain* graphs, we design an algorithm based sampling. Finally, we utilize these *uncertainty*-embedded metaheuristics to effectively selecting edges for further alteration.

C. Uniqueness Score

Intutively, larger amount of noise should be added at nodes that are less anonymized (i.e, more distinctive) in the original graphs. *Uniqueness score* is shown to be an effective metric of how typical is the node v among the nodes of the graph [7]. The formal defintion of uniqueness score is given as follows.

Definition 4. ***Uniqueness Score [7]*** *Let $P : V \rightarrow \Omega_P$ be a property on the set of nodes V of the graph $\mathcal{G}$, let d be a distance function on Ω_P, and let $\theta > 0$ be a parameter. Then the $\theta-commonness$ of the property values $\omega \in \Omega_P$ is $C_\theta(\omega) := \sum_{u \in V} \Phi_{0,\theta}(d(\omega, P(v)))$, while the θ-uniqueness of $\omega \in \Omega_P$ is $U_\theta := \frac{1}{C_\theta(\omega)}$.*

Note that, it adopts a parametric way to estimate the probability density function of the property value ω (i.e., how typical the value is among all the nodes). For the density estimate, we place a normal kernel with standard variance θ which implies the spread out of the property value over the domain. In the previous work [7], they set $\theta = \sigma$, where σ is the standard deviation of the noise generation Gaussian distribution. This is because the larger injected amount of uncertainties indicates that the property values may be spread out in a larger domain.

Here, we set θ equals $\sigma_{\mathcal{G}}$ as the latter represents the spread of property value in the *uncertain graph*. For a given node with the property ω, we can compute its "uniqueness" score as the reverse of its density. The higher the uniqueness score the less-protected the node and the more anonymization it eventually needs.

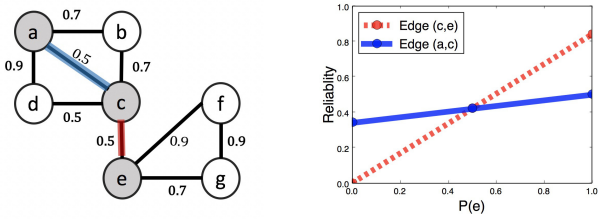
D. Reliability Relevance

Observation When an uncertain edge is altered (partial added or deleted), it will result in the change of reliability. The same amount of alteration performed over different edges may incur significantly different structure change. Referring to the example in Figure 3(a), two vertices a and e will be assigned the same *uniqueness score* due to the exact probabilities associated with their edges. As a result, the *anonymity-oriented* strategy would select and perturb either of the two edges $(\mathtt{a}, \mathtt{c})$ and $(\mathtt{c}, \mathtt{e})$ with the same probability. However, the modifications to $(\mathtt{c}, \mathtt{e})$—which is the only link between two *reliable* clusters—clearly incurs much larger structure distortion than that on $(\mathtt{a}, \mathtt{c})$. Therefore, edge perturbation needs to consider the structural impact of *uncertain* graph edge alteration.

To this end, we propose a theoretically sound estimation for the *"reliability deviation"* caused by individual uncertain edge modifications in a fine-grained way. Following this path, we introduce a new measure, called *Edge Reliability Relevance* ($\mathcal{ERR}$), at the edge level, and an aggregated measure, called *Vertex Reliability Relevance* ($\mathcal{VRR}$), at the vertex level as will be formally defined next. These measures will enable ranking the edges to be targeted for obfuscation in a meaningful utility-based perspective. Besides, we provide an algorithm for computing them efficiently.

Definition 5. ***Two-terminal Reliability Relevance*** *Given an uncertain graph $\mathcal{G}$, and two nodes u and v, the reliability $R_{u,v}(\mathcal{G})$ as defined in Def. 1 is considered as a multivariate function involving all the edge probabilities in $\mathcal{G}$. Thus, given an uncertain edge $e \in \mathcal{G}$, the partial derivative of $R_{u,v}(\mathcal{G})$ w.r.t e's probability variable $p(e)$, denoted as $\mathcal{ERR}^e_{u,v}(\mathcal{G})$, represents the sensitivity of the two-terminal reliability $R_{u,v}$ w.r.t $p(e)$ while all others are held constant. It is defined as:*

$$\mathcal{ERR}^e_{u,v}(\mathcal{G}) = \frac{\partial R_{u,v}(\mathcal{G})}{\partial p(e)}$$



(a) An uncertain graph (b) The reliability $R_{a,e}$ v.s $p(e)$

Figure 3: (a) Different edges have different effect on the overall graph reliability. (b) Formal *Reliability Relevance* measure. A bigger edge's slope indicates big distortion in reliability under small changes to the edge's probability.

Lemma 1. ***Factorization Lemma*** *Given an uncertain graph $\mathcal{G}$, the reliability of the node pair (u, v), i.e., $R_{u,v}(\mathcal{G})$, can be factorized via a specific uncertain edge e as follows:*

$$R_{u,v}(\mathcal{G}) = p(e)R_{u,v}(\mathcal{G}_e) + (1 - p(e))R_{u,v}(\mathcal{G}_{\bar{e}})$$

where uncertain graphs $\mathcal{G}_e$ and $\mathcal{G}_{\bar{e}}$ are identical to the original graph $\mathcal{G}$ with the exception that e is certainly present in the former and certainly not present in the later.

According to the factorization lemma, the partial derivative $\mathcal{ERR}^e_{u,v}$ can be rewritten as:

$$\mathcal{ERR}^e_{u,v}(\mathcal{G}) = R_{u,v}(\mathcal{G}_e) - R_{u,v}(\mathcal{G}_{\bar{e}})$$

On one hand, this factorization indicates that for a given edge e, the incurred reliability discrepancy is **linear** to the amount of edge probability difference. On the other hand, it indicates that edges with different topological locations have different reliability sensitivity. The other crucial point to highlight is that $R_{u,v}(\mathcal{G}_e) - R_{u,v}(\mathcal{G}_{\bar{e}}) \geq 0$ is always true since all connected pairs in $\mathcal{G}_e$ are guaranteed to be a superset or at least equal to that in $\mathcal{G}_{\bar{e}}$.

Considering a single uncertain edge e, the derivatives $\mathcal{ERR}^e_{u,v}(\mathcal{G})$ over all vertex pairs in $\mathcal{G}$ can be arranged in a $|V| \times |V|$ matrix, and as highlighted above, all entires of this matrix equal to or greater than zero. By aggregating these derivatives, we can estimate the overall *reliability relevance* of edge e, denoted as $\mathcal{ERR}^e(\mathcal{G})$, as the sum of all the $\mathcal{ERR}^e_{u,v}$ values. That is:

$$\begin{aligned}\mathcal{ERR}^e(\mathcal{G}) &= \sum_{u,v} |\mathcal{ERR}^e_{u,v}(\mathcal{G})| \\ &= \sum_{u,v} |R_{u,v}(\mathcal{G}_e) - R_{u,v}(\mathcal{G}_{\bar{e}})| \\ &= \sum_{u,v} R_{u,v}(\mathcal{G}_e) - \sum_{u,v} R_{u,v}(\mathcal{G}_{\bar{e}})\end{aligned}$$

Note that $\mathcal{ERR}^e$ equals to the difference of the expected number of connected pairs between the two uncertain graphs $\mathcal{G}_e$ and $\mathcal{G}_{\bar{e}}$ by explicit incorporation of edge uncertainty. In the context of edge relevance, reliability relevance can be seen as generalization of cut-edges, which quantifies the impact of partial edge deletion or addition on the connectivity in the uncertain graph. The higher reliability relevance score of an edge, the bigger impact of edge perturbation over the overall graph.

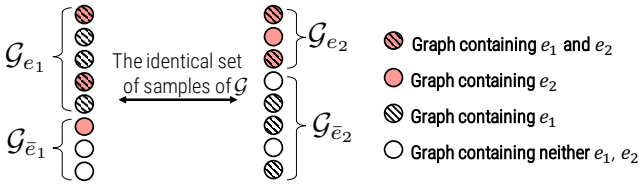


Figure 4: Reused sampling estimator for $\mathcal{ERR}$

On the basis of these edge-level reliablity relevance, we can now compute a vertex-level reliability relevance of a given vertex (Say u) as a weighted sum of reliability relevance of u's edges $\mathtt{E}^u$.

$$\mathcal{VRR}^u(\mathcal{G}) = \sum_{e \in \mathtt{E}^u} p(e)\mathcal{ERR}^e(\mathcal{G})$$

The $\mathcal{VRR}^u(\mathcal{G})$ is a measure of the expected impact of vertex modification on the graph reliability. Namely, the higher the vertex's reliability relevance, the larger reliability distortion introduced by modification associated with its edges.

Reliability Relevance Evaluation Given this theoretical foundation, the challenge is how to evaluate the reliability relevance of edges in a given uncertain graph efficiently ($\mathcal{ERR}-$eval). For each edge e, we need to measure the reliability difference over $\mathcal{G}_e$ and $\mathcal{G}_{\bar{e}}$. This evaluation involves the two-terminal reliability detection problem, which is known to be NP-complete [5].

A baseline algorithm for $\mathcal{ERR}-$eval is to use the Monte Carlo sampling. More precisely, we sample N possible worlds of the input uncertain graph, where N is large enough (around $1,000$) to guarantee high approximation accuracy. Over each sampled possible world (Say G), we carry out a connected-component computation algorithm to count the number of connected pairs $cc(G)$. Then, the count on the original uncertain graph $cc(\mathcal{G})$ can be estimated by taking the average over the sampled deterministic graphs.

Theorem 1. *The complexity of the baseline $\mathcal{ERR}-$eval algorithm is $\mathcal{O}(|E| \cdot N\alpha(|V|)|E|)$ where α is the inverse Ackermann function.*

Proof sketch The time complexity of the connected component detection algorithm based on the union-find method is $\mathcal{O}(\alpha(|V|)|E|)$ [14]. Consequently, computing the $\mathcal{ERR}$ for an edge over the N possible worlds takes time $\mathcal{O}(N\alpha(|V|)|E|)$, and the total time complexity for all the edges is $\mathcal{O}(|E| \cdot N\alpha(|V|)|E|)$.

Obviously, the baseline algorithm is inefficient when the input uncertain graph is very large (it is quadratic in the number of edges). Here, we present a efficient algorithm for $\mathcal{ERR}$ evaluation in Algorithm 2. Its basic idea is to reuse the connected components detection result of samples as illustrated in 4. For each edge e, we group the sampled possible worlds according to the edge existence (Line 4-6), then get the sampled average of cc for each group as accurate approximation of $cc(\mathcal{G}_e)$ and $cc(\mathcal{G}_{\bar{e}})$. By this way, we bring the the evaluation of edge reliability relevance to the realm.

Algorithm 2 Edge Reliability Relevance Evaluation

Input: $\mathcal{G} = (V, E, p)$, N is the number of sampled graphs;
Output: $\mathcal{ERR}$ Reliability relevance of edges in $\mathcal{G}$
1: $CC_e \leftarrow \mathbf{0}$, $CC_{\bar{e}} \leftarrow \mathbf{0}$
2: **for** i=1 **to** N **do**
3: $G \leftarrow$ A deterministic sampled instance
4: $Ind(G)$ is edge existence of sampled graph $\mathcal{G}$
5: $cc(G) \leftarrow$ the number of connected pairs of G
6: $CC_e += Ind(G) \cdot cc(G)$,$CC_{\bar{e}} += (\mathbf{1} - Ind(G)) \cdot cc(G)$
7: **end for**
8: $\mathcal{ERR} = CC_e/p - CC_{\bar{e}}/\mathbf{1} - p$

Theorem 2. *The time complexity of Algorithm2 $\mathcal{ERR}-val$ is $\mathcal{O}(N\alpha(|V|)|E|)$ where N is the number of samples.*

REFERENCES

[1] A. S. Abrahams, E. Coupey, E. X. Zhong, R. Barkhi, and P. S. Manasantivongs. Audience targeting by b-to-b advertisement classification: A neural network approach. *Expert Systems with Applications*, 40(8):2777 – 2791, 2013.

[2] E. Adar and C. Re. Managing uncertainty in social networks. *IEEE Data Eng. Bull.*, 2007.

[3] T. Alsina, D. Wilson, S. Joshi, and S. Sundaresan. Targeting customer segments, Dec. 3 2015. US Patent App. 14/289,118.

[4] S. Asthana, O. King, F. Gibbons, and F. Roth. Predicting protein complex membership using probabilistic network reliability. *Genome research*, 2004.

[5] M. O. Ball. Computational complexity of network reliability analysis: An overview. *IEEE Transactions on Reliability*, 1986.

[6] S. Bhagat, G. Cormode, B. Krishnamurthy, and D. Srivastava. Class-based graph anonymization for social network data. *Proc Vldb Endow*, 2009.

[7] P. Boldi, F. Bonchi, A. Gionis, and T. Tassa. Injecting uncertainty in graphs for identity obfuscation. *SIGMOD*, 2012.

[8] K. Bollacker, C. Evans, P. Paritosh, and T. Sturge. Freebase: a collaboratively created graph database for structuring human knowledge. *SIMMOD*, 2008.

[9] F. Bonchi, A. Gionis, and T. Tassa. Identity obfuscation in graphs through the information theoretic lens. *ICDE*, 2014.

[10] J. Casas-Roma. Privacy-preserving on graphs using randomization and edge-relevance. *Modeling Decisions for Artificial Intelligence*, 2015.

[11] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: user movement in location-based social networks. *kdd*, 2011.

[12] Colbourn and Colbourn. The combinatorics of network reliability. 1987.

[13] S. Das, O. Egecioglu, and A. Abbadi. Anonymizing weighted social network graphs. *ICDE*, pages 904–907, 2010.

[14] M. Fredman and M. Saks. The cell probe complexity of dynamic data structures. *STOC*, 1989.

[15] J. Ghosh, H. Ngo, and S. Yoon. On a routing problem within probabilistic graphs and its application to intermittently connected networks. 2007.

[16] S. Hartung and N. Talmon. The complexity of degree anonymization by graph contractions. *TAMC*, 2015.

[17] M. Hay, G. Miklau, D. Jensen, D. Towsley, and C. Li. Resisting structural re-identification in anonymized social networks. *The VLDB Journal*, 2010.

[18] M. Hay, G. Miklau, D. Jensen, P. Weis, and S. Srivastava. Anonymizing social networks. 2007.

[19] M. Hua and J. Pei. Probabilistic path queries in road networks: traffic uncertainty aware path selection. *EDBT*, 2010.

[20] R. Jin, L. Liu, B. Ding, and H. Wang. Distance-constraint reachability computation in uncertain graphs. *VLDB*, 2011.

[21] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. 2003.

[22] N. Krogan, G. Cagney, H. Yu, G. Zhong, and X. Guo. Global landscape of protein complexes in the yeast saccharomyces cerevisiae. *Nature*, 2006.

[23] J. Lee and C. Clifton. *How Much Is Enough? Choosing ϵ for Differential Privacy*. 2011.

[24] D. Liben Nowell and J. Kleinberg. The link prediction problem for social networks. *The American Society for Information Science and Technology*, 2007.

[25] M. Lin, M. Lin, and R. J. Kauffman. From clickstreams to search-streams: Search network graph evidence from a b2b e-market. *ICEC*, 2012.

[26] K. Liu and E. Terzi. Towards identity anonymization on graphs. *SIGMOD*, 2008.

[27] L. Liu, J. Wang, J. Liu, and J. Zhang. Privacy preservation in social networks with sensitive edge weights. *SDM*, pages 954–965, 2009.

[28] P. Mittal, C. Papamanthou, and D. Song. Preserving link privacy in social network based systems. *NDSS*, 2013.

[29] H. Nguyen, A. Imine, and M. Rusinowitch. Anonymizing social graphs via uncertainty semantics. *CCS*, 2015.

[30] M. Ninggal and J. H. Abawajy. Utility-aware social network graph anonymization. *J Netw Comput Appl*, 2015.

[31] Parchas, Gullo, Papadias, and Bonchi. The pursuit of a good possible world: extracting representative instances of uncertain graphs. *SIGMOD*, 2014.

[32] M. Potamias, F. Bonchi, A. Gionis, and G. Kollios. K-nearest neighbors in uncertain graphs. *VLDB*, 2010.

[33] M. Skarkala, M. Maragoudakis, S. Gritzalis, L. Mitrou, H. Toivonen, and P. Moen. Privacy preservation by k-Anonymization of weighted social networks. *ASONAM*, 2012.

[34] B. Thompson and D. Yao. The union-split algorithm and cluster-based anonymization of social networks. 2009.

[35] Y. Wang, L. Xie, B. Zheng, and K. C. K. Lee. Utility-oriented k-anonymization on social networks. *DASFAA*, 2011.

[36] W. Wu, Y. Xiao, W. Wang, Z. He, and Z. Wang. k-symmetry model for identity anonymization in social networks. *EDBT*, 2010.

[37] X. Ying, K. Pan, X. Wu, and L. Guo. Comparisons of randomization and k-degree anonymization schemes for privacy preserving social network publishing. *SNA-KDD*, 2009.

[38] X. Ying and X. Wu. Randomizing social networks: a spectrum preserving approach. pages 739–750, 2008.

[39] L. Zhang, S. Chen, Y. Jian, Y. Fang, and Z. Mo. Maximizing lifetime vector in wireless sensor networks. *IEEE/ACM Trans. Netw.*, 2013.

[40] B. Zhao, J. Wang, M. Li, F. Wu, and Y. Pan. Detecting protein complexes based on uncertain graph model. *IEEE/ACM Transactions on Computational Biology and Bioinformatics.*, 2014.

[41] B. Zhou and J. Pei. Preserving privacy in social networks against neighborhood attacks. *ICDE*, 2008.