Sharing Uncertain Graphs Using Syntactic 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

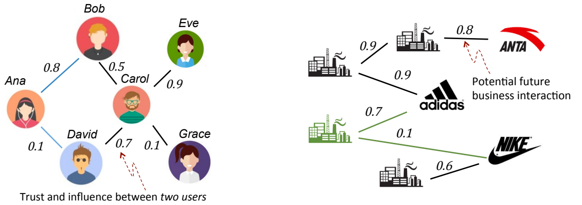
***Abstract*—Many graphs in social and business applications are not deterministic, but are uncertain in nature. Related research requires open access to these uncertain graphs. While sharing these datasets often risks exposing sensitive user data to the public. However, current graph anonymization works only target on deterministic graphs and overlook the uncertain scenario.**

Our work seeks a solution to release uncertain graphs with high utility without compromising user privacy. We show that simply combining the representative extraction strategy and conventional graph anonymization method will result in the addition of noise that significantly disrupts uncertain graph structure. Instead, we introduce an uncertainty-aware method, Chameleon, that provides identical privacy guarantees with much less noise. With the possible world semantics, it enables a fine-grained control over the injected noise. Finally, we apply our method to real uncertain graphs and show that it produces anonymized uncertain graphs that closely match the originals in graph structure statistics.

I. Introduction

In several emerging applications, such as business to business (B2B) and online social networks (OSN), graphs serve as powerful models to capture the inherent complex relationships. 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 as shown in Figure 1. The existence of the edges is inferred with the use of a variety of statistical approaches.

These uncertain graphs are invaluable for scientific research and commercial applications [2, 7]. However, sharing these *uncertain graphs* might violate the privacy of users or entities profiled inside. For example, B2B networks contain information about transactions among companies. Any privacy leak can be used to infer the business model–the key and confidential business asset. It calls for methods to share *uncertain* graphs without compromising privacy.


Bob
Eve
Ana
Carol
David
Grace
0.8
0.5
0.9
0.1
0.7
0.1
Trust and influence between *two users*
0.9
0.8
ANTA
0.9
adidas
Potential future business interaction
0.7
0.1
NIKE
0.6

(a) Social Trust Network

(b) B2B Network

Figure 1: Real-world uncertain graphs with privacy concerns.

State-of-the-Art Techniques. A significant amount of works has been done in privacy preserving deterministic network publishing. Existing methods can be classified into two main categories. The first ones try to publish the data in an anonymized manner. Most of them leverage *syntactic* privacy models derived from k-anonymity to provide ad-hoc privacy protections against specific kinds of attack [9–11, 16, 18]. Another avenue is to apply $\epsilon-$differential privacy for providing the formal privacy guarantee. It roughly falls into two directions. The first direction aims to release differentially private graph mining results such as degree distributions, sub-graph counts, and frequent graph patterns [4]. Such methods limit the use of new queries can be permitted on the data. The second direction aims to publish a sanitized graph [15, 17]. Most research 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 sampling process to generate a fit graph. While, there is no a genearl policy to set the value of ϵ that provides sufficient differential privacy [8].

An obvious approach is to convert uncertain graph anonymization problem into the deterministic scenario by using edge probabilities as edge weights. However, by disregarding possible world semantic of uncertain graphs, such an approach fails to reflect important properties (*i.e.*, connectivity, and spectrum) of uncertain graphs correctly [6, 14].

In summary, these techniques tailored towards deterministic graphs and overlook uncertain ones where each edge carries a degree of uncertainty. The ignorance of edge uncertainty makes these solutions inefficient for sanitizing uncertain graphs. The inefficiency due to various reasons including the wrong assumption of privacy attacks and improper utility loss metrics. In contrast, our approach is able to provide enough privacy guarantee and preserve uncertain graph structure.

Challenges. To achieve our design goals, the following key challenges must be handled.

- *Stochastic Privacy Attacks.* Discarding edge uncertainty in the released graph is impractical since it severely deteriorates data utility. However, the additional release of edge uncertainty makes privacy protection far more difficult. The release of edge uncertainty would empower the adversary and make the profiled entity more vulnerable to stochastic privacy attacks.
- *Stochastic Utility Loss Metric.* It is well-known that the choice of utility loss metric is critical for graph anonymization techniques. Some metrics such as graph edit distance [9], spectrum discrepancy [18], community reconstruction error [16] and shortest path discrepancy [10] are used in prior works.

While, they are heavily tailored for deterministic graphs and built on the top of deterministic graph concepts. Thus, they aren't right choices in the uncertain scenario. We need a well-defined metric to accurately capture the structural deviation of uncertain graphs after anonymization.

• *Intractable Search Space.* Finding an anonymized graph with the desired level of privacy by as few graph contractions as possible is known to be NP-hard [5]. In this work, the edge operation is no longer a binary operation (addition/deletion), but there can be infinite probability values assigned to each edge. It becomes more computationally challenging.

Contribution. Our contributions include:

- We first identify uncertain graph anonymization problem where edge uncertainty needs to be seamlessly integrated into the core of anonymization.
- We propose a benchmark solution, Rep-An, which combines the representative extraction strategy and existing graph anonymization methods together. We show it would deteriorate data utility for edge uncertainty detachment.
- We develop a new scheme, Chameleon, which adopts several stochastic shifts including the evaluation of privacy risk, utility loss, and judicious stochastic modifications following the possible world semantic.
- Experimental study on real-world datasets shows the significant improvement of Chameleon over Rep-An.

II. Problem Definition

A. Uncertain Graph

According to the possible-world semantics [3], an uncertain graph $\mathcal{G} = (V, E, p)$ essentially represents a probability distribution over all of the certain graphs G in the forms of which the uncertain graph may actually exist. The probability of observing any possible world $G_i = (V, E_{G_i})$ is [1]

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

B. Reliability Discrepancy

Prior works show the connectivity model is able to yield a better graph representation than degree sequence model [10]. Besides, connectivity discrepancy was proven to be a proper utility-loss metric in graph anonymization works. Inspired by these works, we use its generalized version – Reliability Discrepancy (RD) as the utility-loss metric in the uncertain scenario. Reliability $R_{u,v}(\mathcal{G})$ captures the probability that two given (sets of) nodes are reachable over all possible worlds of the uncertain graph as follows:

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

where $\mathcal{I}_G(u, v)$ is 1 iff u and v are connected in G, and 0 otherwise. Naturally, reliability discrepancy of a anonymized uncertain graph $\tilde{\mathcal{G}}$ w.r.t. the original one $\mathcal{G}$, is defined as the sum of the reliability deviation over all node pairs.

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

C. Privacy Policy

We adopt the (k, ϵ)-obf privacy model introduced by Boldi et al. [1], where $k \geq 1$ is a desired level of obfuscation and $\epsilon \geq 0$ is a tolerance parameter. Analogous to $k-$anonymity, $k-$obf requires blending every node with other *fuzzy* match nodes. The level of *fuzzy* matching is quantified by the entropy over posterior probabilities. The stochastic nature makes it a good fit in the uncertain graph context. Moreover, the introduction of tolerance parameter ϵ makes it suitable for real practice.[2]

Definition 1. ***(k, ϵ)-obf*** *Let P be a vertex property (i.e., vertex degree in our work), 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 the original* uncertain graph *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 vertex 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 our goal.

Problem 1. *Given an uncertain graph $\mathcal{G}$ and desired anonymization parameters k and ϵ, the objective is to find a (k, ϵ)-obf one $\tilde{\mathcal{G}} = (V, E, \tilde{p})$ with minimal $\Delta(\tilde{\mathcal{G}})$ as*

$$\underset{\tilde{\mathcal{G}}}{\operatorname{argmin}} \quad \Delta(\tilde{\mathcal{G}})$$

$$\textit{Subject to} \quad \tilde{\mathcal{G}} \textit{ is } (k, \epsilon) - \textit{obf}$$

III. Uncertain Graph Anonymization

A. Benchmark Solution

Inspired by the methodology "uncertain graph processing through representative instances", we propose the benchmark solution Rep-An: a given *uncertain graph* $\mathcal{G}$ is approximated as a single *representative* (deterministic) graph G_{rep}, then outputs the *anonymized* result of G_{rep} as the final result.

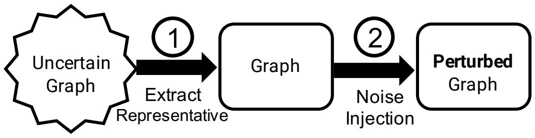


Figure 2: Overview of Rep-An.

By this way, the state-of-art methods can be applied on *uncertain graphs*, regardless of the inherent uncertainty. It is the first time to combine isolated works (representative extraction [13] and graph anonymization [1]) together for uncertain graph anonymization.

However, this approach is problematic. First, the input edge uncertainties (probabilities) are no longer integrated into the anonymization process since they are detached in the first step. Second, since the two phases are isolated from each other, different phases might be optimized for different metrics. As the result, Rep-An might introduce a high level of noise and consequently deteriorate the overall data utility.

[1] In this work, we assume that the existence probabilities of edges are *mutually independent*. Different uncertainty models will be considered later.

[2] There might be extreme unique nodes, e.g., Trump in a Twitter network, whose obfuscation is almost impossible.

B. The Chameleon Algorithm

Instead of detaching edge uncertainty from the anonymization phase, we shift the state-of-art method [1] by integrating uncertainty semantics into its core steps, namely Chameleon. It enables a unifying and grained control over the noise injected to *uncertain graphs*, then provides enough privacy guarantee with better utility.

Problem Transformation Anonymization is done via altering the probabilities of sampled edges. For each sampled edge e, it is assigned a probability deviation r_e, where $r_e \leftarrow R(\sigma)$[3]. As the standard deviation σ decreases, a greater mass of R_σ will concentrate near $r_e = 0$, then the amount of injected noise and consequent structural deviation will be smaller. It enables us to transform the graph anonymization problem into the minimization of structural noise need to be (k, ϵ)-*obf*. The computation of the later one can be achieved via a binary search on the value of standard deviation σ.

Search Flow The binary search flow is determined by the GenObf function. For the given standard deviation σ, Genobf either returns the best found (k, ϵ)-*obf* instance or indicates failure. The anonymization algorithm starts with an initial guess of an upper bound σ_u which is iteratively doubled until a (k, ϵ)-*obf* graph is found. Then, the binary-search process is performed using σ_l = 0 as the lower bound, and the found upper bound σ_u. The binary search terminates when the search interval is sufficiently short. It outputs the best (k, ϵ)-*obf* found (i.e., the last one that was successfully generated).

GenObf To find a (k, ϵ)-*obf* of the input uncertain graph $\mathcal{G}$ using a given standard deviation σ, Genobf performs t randomized attempts (In our experiment, we used $t = 5$). Iff all t attempts fail, Genobf returns failure signal. Otherwise, it returns the (k, ϵ)-*obf* with minimal ϵ. Each attempt begins by ***selecting*** a subset E_c, which will be subjected to alteration [4]. Next, we ***distribute*** the deviation among selected edges and ***alter*** their edge probabilities. For more detail of the search, we refer interested readers to the state-of-art work [1] since Chameleon ensembles it except the design of heuristic and the edge perturbation scheme.

C. Heuristic Search inside Chameleon

Since the number of possible *anonymized* graphs is exponentially large, finding the optimal one is NP-hard. Thus, the heuristic search is commonly employed to find a reasonably good one [1, 12].

Deterministic Uniqueness In previous work [1], the perturbation r_e for each edge $e = (u, v)$ is assigned according to the uniqueness score of the vertices u and v. Intuitively, a larger perturbation is assigned around more distinctive vertices w.r.t vertex property P. More specifically, the commonness (the inverse of uniqueness) is used to measure the typical level of the property value $P(v)$. It amounts to the weighted average *distance* among all other property values. Note that, all the property values are scalar. While the heuristic has been proved to be successful, it heavily tailored towards deterministic graphs and implicitly assumes that uniform edge relevance.

Stochastic Uniqueness We extend the preliminary version of uniqueness score for handing stochastic cases. Most notably, we represent the vertex property value ρ as a parameterized probability distribution and use KL-divergence to measure the distance between two probability distributions. The *stochastic commonness* $\mathcal{SC}$ (the inverse of stochastic uniqueness $\mathcal{SU}$) of a given property value $\rho_v = P(v)$ is given by

$$\mathcal{SC}(\rho_v) = \sum\nolimits_{u \in V} \Phi_{0,\theta}(D_{KL}(\rho_v || \rho_u))$$

where the weight decays exponentially as a function of the distance between ρ_v and ρ_u and the parameter θ determines the decay rate. We set $\theta = \sigma$ as the injected noise blurs the meta distribution of property values.

Edge Reliability Relevance Alteration over a single edge would produce structural change and send ripples through the rest of the graph. We observe that the same amount of deviation r_e assigned over different edges will incur significantly different structural changes. Clearly, the uniform edge relevance assumption does not hold. Instead of penalizing edge alteration uniformly, we should penalize it according to edge relevance. Therefore, edge relevance ($\mathcal{ER}$) needs to be measured. There are many potential ways to measure it. Importantly, this measure must be fitted to the problem.

In this work, we measure the introduced structural deviation by a ***unit*** r_e, edge reliability relevance ($\mathcal{ERR}(e)$), as

$$\mathcal{ERR}(e) = \lim_{r_e \to 0} \frac{\Delta(\mathcal{G} + r_e)}{r_e} = \sum_{u,v} \frac{|R_{u,v}(\mathcal{G} + r_e) - R_{u,v}(\mathcal{G})|}{r_e}$$

where the factorization lemma [6] indicates

$$R_{u,v}(\mathcal{G} + r_e) - R_{u,v}(\mathcal{G}) = r_e \cdot \left[R_{u,v}(\mathcal{G}_e) - R_{u,v}(\mathcal{G}_{\hat{e}})\right]$$

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

Clearly, $\mathcal{ERR}(e)$ amounts to the number difference of connected node pairs between two neighbor uncertain graphs $\mathcal{G}_e$ and $\mathcal{G}_{\bar{e}}$, where $\mathcal{G}_e$ and $\mathcal{G}_{\bar{e}}$ are identical to $\mathcal{G}$ with the exception that $p(e) = 1$ in the former and $p(e) = 0$ in the later.

Intuitively, $\mathcal{ERR}$ generalizes the concept of the cut-edge by quantifying the stochastic impact of edge alteration over the connectivity of all the possible worlds. We also show its computation can be achieved via Monte-Carlo sampling with an acceptable time complexity $\mathcal{O}(N \cdot T)$, where N is the number of samples and T is time complexity for the employed connected component detection algorithm. [5] The key idea is to memorize the calculated result (# of connected node pairs) over samples. For each edge e, we group then aggregate results of samples according to edge existence. And, the vertex reliability relevance $\mathcal{VRR}$ amounts to the weighted sum of $\mathcal{ERR}$ values.

Combination We believe that the combination of stochastic uniqueness $\mathcal{SU}$ and reliability relevance $\mathcal{RR}$ heuristic would introduce cumulative benefit in heuristic search since they are

[3] Following [1], the distribution $R(\sigma)$ is a truncated normal distribution with mean 0 and variance σ^2, but could in principle be any distribution.

[4] The set E_c, whose target size is $|E_c| = c|E|$

[5] The union-and-find method is used in this work.

Table I: Characteristics of the datasets and privacy parameters

Graph	Content	Nodes	Edges	Edge Prob	ϵ
PPI	Protein-Protein Interaction	12K	397K	0.29	10^{-2}
BK	Location-based OSN	58K	214K	0.29	10^{-3}
DBLP	Co-authorship Network	824K	5M	0.46	10^{-4}

complementary to each other. The resulting search heuristic is their convex combination: $\mathcal{SU} - \beta \cdot \mathcal{VRR}$. In practice, we found that $\beta = 1$ worked best. Intuitively, we prioritize the perturbation over more distinctive nodes and penalize the perturbation over "structural relevant" nodes.

D. Stochastic Controlled Perturbation inside Chameleon

For each sampled edge e with the distributed standard deviation σ_e, we select probability deviation r_e from the random distribution $R(\sigma_e)$. Thus, we are left asking the question, *how can we safely alter edge probability for higher anonymity?* It is quite straightforward in the deterministic scenario [1]:

$$\tilde{\rho}(e) := \begin{cases} 1 - r_e & \rho(e) = 1 \\ r_e & \rho(e) = 0 \end{cases}$$

Following the possible world semantics, we propose the general version as $\tilde{\rho}(e) := \rho(e) \cdot [1 - r_e] + (1 - \rho(e)) \cdot [r_e]$. It enables stochastic control $\mathcal{SC}$ over the amount of injected noise and limits the range of the probability value $\tilde{\rho}(e)$ to the range $[\rho(e), 1 - \rho(e)]$ instead of the wider one $[0, 1]$. And, we theoretically show it follows a likelihood gradient which maximizes the overall entropy (disorder), an approximation of anonymity of the input uncertain graph $\mathcal{G}$.

IV. Experiments

A. Settings

We compare our proposed scheme Chameleon with three baselines. The first one is Rep-An which anonymizes an uncertain graph through the representative instance. The second is its variant (-RR) which ignores edge reliability relevance in heuristic search. The third is another variant (-SC) without the use of stochastic controlled edge perturbation strategy.

We test them on three uncertain graphs: PPI, BrightKite (BK) and DBLP as described in Table I. We consider the obfuscation level k in the range $[100, 300]$, and possible tolerance value ϵ to explore their performance differences.

For every obfuscated graph, we sampled 1000 possible worlds to compute its statistics of interest: average node degree, degree distribution, average distance, graph diameter, reliability, and clustering coefficient. Here, we report their reliability discrepancies (RD) against the original one. The smaller RD, the better uncertain graph structure preserving.

B. Results

Chameleon vs. Rep-An Figure 3 shows the performance of Chameleon is always much better than Rep-An. As the size of graph increases (PPI→ DBLP), the performance gap becomes larger and larger. Even in the case $k = 100$ (weak privacy guarantee with little noise requirement), Rep-An still introduces considerable structural distortion. We believe that structural distortion can be largely attributed to the detachment of edge uncertainty. The representative instance extraction step might introduce too much noise. It also results in cumulative errors in the anonymization step. Consequently, the generated anonymized outputs are significantly different from the original uncertain graph, as witnessed by Figure 3.

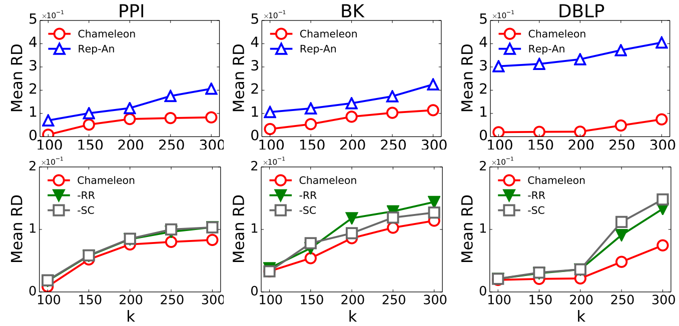


Figure 3: Comparison of different anonymization algorithms.

Chameleon vs. Variants In all the cases, Chameleon outperforms its variants. When we increase the strength of privacy guarantee k, the introduced RD (structural distortion) progressively increases. While the RD introduced by Chameleon increases relatively slowly, within 10%. Thus, we can safely conclude that RR and SC both contribute to the fine-grained control of noise, and lead to better data utility.

V. Conclusions and Future Work

In this work, we first identify the overlooked problem–uncertain graph anonymization. Then, we develop a new scheme, Chameleon, which seamlessly integrates edge uncertainty into the core of the anonymization process such as the evaluation of privacy risk, utility loss, and judicious stochastic modifications. Experiments on three real-world datasets verify its effectiveness. In real-world graphs, edge probabilities sometimes are not independent, but dependent. Thus, we plan to extend our scheme to handle uncertain graphs with dependent probabilities. It is also interesting to investigate sharing uncertain graphs in the differentially private manner.

References

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

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

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

[4] W.-Y. Day, N. Li, and M. Lyu. Publishing graph degree distribution with node differential privacy. *SIGMOD*, 2016.

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

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

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

[8] J. Lee and C. Clifton. How much is enough? choosing ϵ for differential privacy. *ISC*, 2011.

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

[10] L. Liu, J. Wang, J. Liu, and J. Zhang. Privacy preservation in social networks with sensitive edge weights. *SDM*, 2009.

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

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

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

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

[15] A. Sala, X. Zhao, C. Wilson, and H. Zheng. Sharing graphs using differentially private graph models. *IMC*, 2011.

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

[17] Q. Xiao, R. Chen, and K. Tan. Differentially private network data release via structural inference. *KDD*, 2014.

[18] X. Ying and X. Wu. Randomizing social networks: a spectrum preserving approach. *SIAM*, 2008.