DK-MICROAGGREGATION: ANONYMIZING GRAPHS WITH DIFFERENTIAL PRIVACY GUARANTEES

Masooma Iftikhar
Qing Wang
Yu Lin

Research School of Computer Science
The Australian National University

May 13, 2020
OUTLINE

- Introduction
- Problem Formulation
- dK-Microaggregation Framework
- Proposed Algorithm
- Experiments and Results
- Conclusion and Future Work
INTRODUCTION
Graph data analysis has been widely performed in real-life applications. For instance,

- online social networks are explored to analyze human social relationships;
- election networks are studied to discover different opinions in a community.

However, such networks often contain sensitive or personally identifiable information, such as social contacts, personal opinions and private communication records.

Publishing graph data can thus pose a privacy threat.
**Graph Data Release Process**

*Figure 1: Graph Data Release Process (e.g. online social network)*
Aims and Challenges

■ **Aim:** To generate anonymized graphs with $\varepsilon$-differential privacy guarantee for improving utility of anonymized graphs being published.

■ **Key Challenges:**
  - To preserve topological structures of an original graph at different levels of granularity.
  - To enhance utility of graph data by reducing the magnitude of noise needed to achieve $\varepsilon$-differential privacy through adding controlled perturbation to its edges (i.e., edge privacy).

■ **Key Observation:** We observe that the dK-graph model [5] for analyzing network topologies can serve as a good basis for generating structure-preserving anonymized graphs.
Problem Formulation
The dK-graph model [5] provides a systematic way of extracting subgraph degree distributions from a given graph, i.e. *dK-distributions*.

A *dK-distribution* $dK(G)$ over a graph $G$ is the probability distribution on the connected subgraphs of size $d$ in $G$.

Specifically, *1K-distribution* captures a degree distribution, and *2K-distribution* captures a joint degree distribution. When $d = |V|$, dK-distribution specifies the entire graph.

A *dK-distribution* is extracted from a graph, by using *dK function* (s.t. $\gamma^{dK}(G) = dK(G)$).
We define $dK$-graph as a graph that can be constructed through reproducing the corresponding $dK$-distribution.

A $dK$-graph over $dK(G)$ is a graph in which connected subgraphs of size $d$ satisfy the probability distribution in $dK(G)$.

Conceptually, a $dK$-graph is considered as an anonymized version of an original graph $G$ that retains certain topological properties of $G$ at a chosen level of granularity.

We aim to generate $dK$-graphs with $\varepsilon$-differential privacy guarantee for preserving privacy of structural information between nodes of a graph (edge privacy).
**Problem Statement**

- Two graphs $G = (V, E)$ and $G' = (V', E')$ are said to be **neighboring graphs**, denoted as $G \sim G'$, iff $V = V'$, $E \subseteq E'$ and $|E| + 1 = |E'|$.

**Differentially private dK-graphs**

A randomized mechanism $\mathcal{K}$ provides $\varepsilon$-differentially private dK-graphs, if for each pair of neighboring graphs $G \sim G'$ and all possible outputs $G \subseteq \text{range}(\mathcal{K})$, the following holds

$$
\Pr[\mathcal{K}(G) \in G] \leq e^{\varepsilon} \times \Pr[\mathcal{K}(G') \in G].
$$

- $\mathcal{G}$ is a family of dK-graphs, and $\varepsilon > 0$ is the **differential privacy parameter**. Smaller values of $\varepsilon$ provide stronger privacy guarantees.
dK-Microaggregation Framework
We incorporate microaggregation techniques [1] into the dK-graph model [5] to reduce the amount of random noise without compromising $\varepsilon$-differential privacy.

Generally, dK-microaggregation works in the following steps:

1. Extracts a dK-distribution from each neighboring graph.
2. Microaggregates the dK-distribution and perturbs the microaggregated dK-distribution to generate $\varepsilon$-differentially private dK-distribution.
3. Generates $\varepsilon$-differentially private dK-graphs using a dK-graph generator [4, 5].
Figure 2: A high-level overview of the proposed framework (dK-Microaggregation).
PROPOSED ALGORITHM
A microaggregation algorithm for dK-distributions $M = (C, A)$ consists of two phases:

(a) **Partition** - similar tuples in a dK-distribution are partitioned into the same cluster;

(b) **Aggregation** - the frequency values of tuples in the same cluster are aggregated.

**Figure 3**: An illustration of our proposed algorithms.
**Proposed Microaggregation Algorithms**

- **MDAV-dK algorithm**: We use a simple microaggregation heuristic, called *Maximum Distance to Average Vector (MDAV)* [1], which can generate clusters of the same size $k$, except one cluster of size between $k$ and $2k - 1$. Then unlike MDAV, we aggregate frequency values of tuples in each cluster. However, **MDAV-dK** would suffer significant information loss when evenly partitioning a highly skewed dK- distribution into clusters of the same size.

- **MPDC-dK algorithm**: To address this issue, we propose *Maximum Pairwise Distance Constraint (MPDC-dK)*, which aims to partition a dK-distribution into a minimum number of clusters in which every pair of tuples from the same cluster satisfies a distance constraint $\tau$. 
EXPERIMENTS AND RESULTS
Experimental Setup

Three network datasets:

1. *polbooks* contains 105 nodes and 441 edges.
2. *ca-GrQc* contains 5,242 nodes and 14,496 edges.
3. *ca-HepTh* contains 9,877 nodes and 25,998 edges.

Two measures:

- **Euclidean distance** [6] measures network structural error between original and perturbed dK-distributions.
- **Sum of absolute error** [2] measures within-cluster homogeneity of clustering algorithms, defined as:

  \[
  SAE = \sum_{i=1}^{N} \sum_{\forall x_j \in c_i} |x_j - \mu_i|,
  \]

  where \( c_i \) is the set of tuples in cluster \( i \) and \( \mu_i \) is the mean of cluster \( i \).
To verify the utility, we compare the structural error between original and perturbed dK-distributions generated by MDAV-dK, MPDC-dK and the baseline method $\varepsilon$-DP. Our proposed algorithms MDAV-dK and MPDC-dK lead to less structural error for every value of $\varepsilon$ as compared to $\varepsilon$-DP.
We compare the quality of clusters, in terms of within-cluster homogeneity, generated by MDAV-dK and MPDC-dK. MPDC-dK outperforms MDAV-dK by producing clusters with less SAE over all three datasets.

### Table 1. Performance of MDAV-dK under different values of $k$.

| Datasets | Measures | $k=1$ | $k=3$ | $k=5$ | $k=7$ | $k=9$ | $k=11$ | $k=13$ | $k=15$ |
|----------|----------|-------|-------|-------|-------|-------|-------|-------|-------|
| polbooks | SAE      | 0     | 144.6 | 184.67| 224.84| 273.6 | 292.21| 299.15| 334.25|
|          | # Clusters | 161   | 53    | 32    | 23    | 17    | 14    | 12    | 10    |
| ca-GrQc  | SAE      | 0     | 1073.3| 1476  | 1810.5| 2166.8| 2313.7| 2555.5| 2730  |
|          | # Clusters | 1233  | 411   | 246   | 176   | 137   | 112   | 94    | 82    |
| ca-HepTh | SAE      | 0     | 968.72| 1304  | 1599.8| 1893.9| 2063  | 2232.9| 2389.7|
|          | # Clusters | 1295  | 431   | 259   | 185   | 143   | 117   | 99    | 86    |

### Table 2. Performance of MPDC-dK under different values of $\tau$.

| Datasets | Measures | $\tau=1$ | $\tau=3$ | $\tau=5$ | $\tau=7$ | $\tau=9$ | $\tau=11$ | $\tau=13$ | $\tau=15$ |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| polbooks | SAE      | 90.72    | 192.15   | 328.96   | 424.2    | 563.73   | 617.63   | 723.06   | 795.77   |
|          | # Clusters | 68      | 25       | 13       | 8        | 7        | 5        | 3        | 3        |
| ca-GrQc  | SAE      | 725.38   | 1732.1   | 2630.6   | 3470.6   | 4262.9   | 5176.7   | 6170.1   | 7037.7   |
|          | # Clusters | 483     | 178      | 98       | 61       | 42       | 35       | 26       | 20       |
| ca-HepTh | SAE      | 841.87   | 1761.8   | 2773.3   | 3721.4   | 4719.2   | 5623.8   | 6402.6   | 7034.2   |
|          | # Clusters | 412     | 140      | 73       | 37       | 34       | 24       | 19       | 15       |
CONCLUSION AND FUTURE WORK
Conclusion:

► We present a novel framework, called *dK-microaggregation*, that can leverage a series of network topology properties to generate \( \varepsilon \)-differentially private anonymized graphs.

► We propose a *distance constrained algorithm* for approximating dK-distributions of a graph via microaggregation within the proposed framework, which can reduce the amount of noise being added into \( \varepsilon \)-differentially private anonymized graphs.

► The effectiveness of our proposed framework has been empirically verified over three real-world network.

Future work: To this work will consider zero knowledge privacy (ZKP) [3], to release statistics about social groups in a network while protecting privacy of individuals.
REFERENCES

Josep Domingo-Ferrer and Vicenç Torra.
**Ordinal, continuous and heterogeneous k-anonymity through microaggregation.**
Data Mining and Knowledge Discovery, 11(2):195–212, 2005.

Vladimir Estivill-Castro and Jianhua Yang.
**Fast and robust general purpose clustering algorithms.**
In PRICAI, pages 208–218, 2000.

Johannes Gehrke, Edward Lui, and Rafael Pass.
**Towards privacy for social networks: A zero-knowledge based definition of privacy.**
In TCC, pages 432–449, 2011.

Priya Mahadevan, Calvin Hubble, Dmitri Krioukov, Bradley Huffaker, and Amin Vahdat.
**Orbis: Rescaling degree correlations to generate annotated Internet topologies.**
In SIGCOMM, pages 325–336, 2007.

Priya Mahadevan, Dmitri Krioukov, Kevin Fall, and Amin Vahdat.
**Systematic topology analysis and generation using degree correlations.**
In SIGCOMM, pages 135–146, 2006.

Alessandra Sala, Xiaohan Zhao, Christo Wilson, Haitao Zheng, and Ben Y Zhao.
**Sharing graphs using differentially private graph models.**
In SIGCOMM, pages 81–98, 2011.
Thanks for your attention!

Any questions?