An Anonymiser Tool for Sensitive Graph Data

Charini Nanayakkara, Peter Christen and Thilina Ranbaduge

Research School of Computer Science, The Australian National University, Canberra, ACT 2600 Australia

Abstract

Analysis of graph data is extensively conducted in numerous domains to learn the relationships between and behaviour of connected entities. Many graphs contain sensitive data, for example social network users and their posts, or genealogical records such as birth and death certificates. This has limited the use and publication of such sensitive graph data sets. While there are various techniques available to anonymise tabular data, anonymising graph data while maintaining the node and edge structure of the original graph, such as node attributes and the similarities between nodes, is a challenging task. In this paper, we present a web tool which can anonymise sensitive graph data while maintaining the similarity structure of the original graph by employing a cluster-based mapping of sensitive to public attribute values, as well as randomly shifting date values. Our demonstration will illustrate the tool on two example data sets of historical birth records.

Keywords

Graph anonymisation, sensitive data, data privacy, data generation, cluster mapping, string similarity

1. Introduction

Representing databases as graphs is often necessary in modern data analysis tasks due to many databases having inter-relationships between their records. For example, a social network data set would represent individuals as nodes and their relationships as edges, whereas several census data sets may be connected with edges to show the records that potentially belong to members of the same family (such as siblings). The types of data which require a graph representation are often sensitive (such as data that represent real people) and therefore cannot be shared publicly. This requires the anonymisation of a graph such that sensitive data cannot be reidentified, while the structure of the graph, the relationships between nodes and their attributes, are being preserved.

Anonymisation of graph data is a topic that has been explored by several previous studies [1, 2]. However, these studies focus on protecting a data set against privacy attacks [3, 4] and therefore often compromise the structure of the original graph by removing or adding edges and/or nodes that are vulnerable to reidentification due to their unique characteristics. Furthermore, the data sets resulting from existing anonymisation techniques do not necessarily have to be interpretable by humans. Rather, the aim of these techniques is to anonymise a graph such that identifying the real-world entities represented by nodes in the graph is made difficult, while it is still possible to conduct analysis on the anonymised graph with suitable machine learning algorithms.

Therefore, existing graph anonymisation tools are not useful in generating an anonymised, human interpretable version of a given sensitive graphdata set. An anonymised human interpretable data set is however important to allow inspection of the data set in the context of transparency of how a machine learning algorithm performs on that data set, or to be able to publish a data set for educational purposes.

In this paper, we present our web tool DOYEN (**D**ata an**O**n**Y**miser for s**EN**sitive Graph Data) which can generate an anonymised version of a sensitive graph data set while maintaining its graph structure. Our method replaces the sensitive attribute values of nodes with values from public lookup tables using a cluster based mapping technique. The initial implementation of the DOYEN tool anonymises family data where graph connectivity represents sibling relationships. Such family data are required for numerous social science studies and for the reconstruction of (historical) populations [5].

We anticipate that DOYEN would be instrumental in mitigating hindrances to such research work due to the inability of publishing sensitive graph data. We next provide a brief overview of the DOYEN system. In Sect. 3 we describe the cluster-based mapping, and in Sect. 4 the process of generating anonymised data. We then discuss in Sect. 5 the demonstration of DOYEN and illustrate its front end.

Proceedings of the CIKM 2020 Workshops, October 19-20, 2020, Galway, Ireland

EMAIL: charini.nanayakkara@anu.edu.au (C. Nanayakkara); peter.christen@anu.edu.au (P. Christen); thilina.ranbaduge@anu.edu.au (T. Ranbaduge)
ORCID: 0000-0002-7603-1845 (C. Nanayakkara); 0000-0003-3435-2015 (P. Christen); 0000-0001-5405-3704 (T. Ranbaduge)

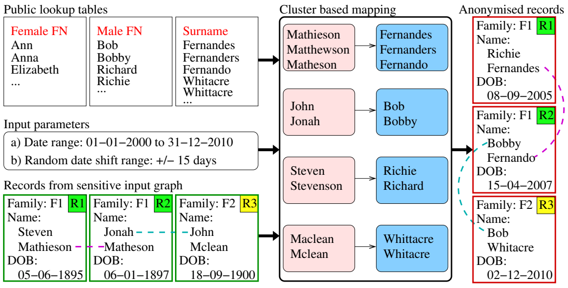


Figure 1: Overview of the DOYEN system to generate anonymised graph data.

Algorithm 1: ***Cluster based attribute value mapping***

```
Input:
- $\mathbf{D}^s$:  Sensitive input data set                - A: Set of sensitive attributes
- L:     Attribute value lookup tables

Output:
- M:   Attribute value mapping table

1: M = { },        // Initialise empty attribute value mapping table
2: for $a_i \in$ A do:                // Iterate over sensitive attributes
3:     $\mathbf{C}_d$ = GetCluster($\mathbf{D}^s.\mathbf{a}_i$)      // Cluster input attribute values
4:     $\mathbf{C}_l$ = GetCluster($\mathbf{L}.\mathbf{a}_i$)      // Cluster lookup attribute values
5:     for $\mathbf{c}_j \in \mathbf{C}_d$ do:        // Iterate over attribute value clusters
6:         $\mathbf{c}'_j$ = BestMatch($\mathbf{c}_j$.s, $\mathbf{c}_j$.l, $|\mathbf{c}_j.\mathbf{v}|$, $\mathbf{C}_l$)        // Find best match
7:         $\mathbf{C}_l.remove(\mathbf{c}'_j)$ // Remove selected cluster from lookup clusters
8:         MapValues(M, $\mathbf{c}_j$.v, $\mathbf{c}'_j$.v)   // Map attribute values in clusters
9: return M
```

2. System Overview

The DOYEN tool anonymises a given sensitive input graph data set by replacing sensitive attribute values with values from public lookup tables. At the same time DOYEN maintains the structure of the graph and the similarities between its nodes by conducting attribute value replacement in a way that preserves the similarities between nodes.

Figure 1 illustrates a high level overview of DOYEN. The sensitive graph data set and one or more lookup tables with attribute values extracted from a public data source are provided as input to DOYEN. As example we show a family graph data set where siblings have the same family ID and colour, and highly similar first name and last name pairs (where similarity is calculated with an approximate string similarity function [6]) are shown as edges (dashed lines). DOYEN first clusters the sensitive attribute values from the input graph data set and each public lookup table separately, and then maps the generated clusters of sensitive values to the clusters generated from a lookup table using a mapping approach as we describe below.

For each node in the input graph data set an anonymised node is then generated by replacing each sensitive attribute value with the corresponding mapped value from a public lookup table. If the graph data contains date values, as shown in our example in Fig. 1, they are anonymised by shifting dates within a user specified range as we discuss in Sect 4.

3. Cluster based Attribute Value Mapping

We now describe our anonymisation approach for sensitive attribute values. Assume we have a sensitive input data set $\mathbf{D}^s$ containing records $r \in \mathbf{D}^s$ that represent entities, and external lookup tables $\mathbf{L}$ of attribute values. The data set $\mathbf{D}^s$ can be represented as a graph $\mathbf{G}^s = (\mathbf{V}^s, \mathbf{E}^s)$, where a node (vertex) in $\mathbf{V}^s$ represents a record $r_i \in \mathbf{D}^s$, and an edge in $\mathbf{E}^s$ corresponds to the pairwise attribute similarity of the record pair (r_i, r_j). Such similarities, as calculated by comparing attribute values, are often used to show the strength or importance of relationships in graph data [7]. We refer to the set of sensitive attributes in $\mathbf{D}^s$ as $\mathbf{A} = \{a_1, \dots, a_n\}$, and the values from each sensitive attribute a_i in the input data set and the lookup tables as $\mathbf{D}^s.\mathbf{a}_i$ and $\mathbf{L}.\mathbf{a}_i$, respectively.

Assuming that the sensitive attributes $\mathbf{A}$ have been used to calculate the pairwise similarities between records in $\mathbf{D}^s$, we need to ensure that these similarities are maintained in the anonymised data set we generate. This means that we need to retain the similarity structure of $\mathbf{G}^s = (\mathbf{V}^s, \mathbf{E}^s)$ in the generated anonymised graph $\mathbf{G}^a = (\mathbf{V}^a, \mathbf{E}^a)$ which represents the anonymised data set $\mathbf{D}^a$. To achieve this goal, we use a one-to-one cluster mapping approach where we map an attribute value cluster from the sensitive input data set $\mathbf{D}^s$ to an attribute value cluster from the public lookup tables $\mathbf{L}$ such that the intra cluster similarities are highly similar across the two clusters.

Algorithm 1 outlines this approach to anonymise the sensitive attribute values in a given graph data set. The input to the algorithm are the set of attributes $\mathbf{A}$ (such as names and addresses of people), the sensitive graph data set $\mathbf{D}^s$, and the lookup tables $\mathbf{L}$ which contain values that attributes $a_i \in \mathbf{A}$ could assume.

In lines 2 to 4, the algorithm iterates over the sensitive attributes $a_i \in \mathbf{A}$, and clusters the corresponding attribute values in the input data set $\mathbf{D}^s.\mathbf{a}_i$ and the lookup table $\mathbf{L}.\mathbf{a}_i$. Next, in lines 5 and 6, we iterate over the attribute value clusters $\mathbf{c}_j \in \mathbf{C}_d$ from the input data set $\mathbf{D}^s$ and find the best matching attribute value cluster from the lookup attribute value clusters $\mathbf{C}_l$. For a given attribute value cluster $\mathbf{c}_j$, we obtain its sorted values $\mathbf{c}_j.\mathbf{v} = [v_1, v_2, \dots v_m]$, the vector of pairwise similarities of attribute values in cluster $\mathbf{c}_j.\mathbf{s} = [s_{v_1,v_2}, \ldots, s_{v_1,v_m}, s_{v_2,v_3}, \ldots, s_{v_{m-1},v_m}]$, and the attribute value length vector $\mathbf{c}_j.\mathbf{l} = [|v_1|, |v_2|, \ldots, |v_m|]$. The best matching cluster for $\mathbf{c}_j$ is identified with the function **BestMatch**(), which takes as input the pairwise similarities vector $\mathbf{c}_j.\mathbf{s}$, the vector of attribute value lengths $\mathbf{c}_j.\mathbf{l}$, the number of attribute values in cluster $|\mathbf{c}_j.\mathbf{v}|$, and the set of lookup attribute value clusters $\mathbf{C}_l$. The function **BestMatch**() finds the most similar lookup cluster $\mathbf{c}'_j$ to $\mathbf{c}_j$ from the set of lookup clusters $\mathbf{C}_l$ (which are of the same size as $\mathbf{c}_j$) using Euclidean distances calculated between their similarity vectors and their attribute value length vectors. The cluster $\mathbf{c}'_j \in \mathbf{C}_l$ with the minimal weighted distances is selected to be mapped to cluster $\mathbf{c}_j$. If $\mathbf{C}_l$ does not contain clusters of size $|\mathbf{c}_j.\mathbf{v}|$ then subsets of clusters from $\mathbf{C}_l$ which are larger than $|\mathbf{c}_j.\mathbf{v}|$ are considered.

In line 7, we remove the selected cluster $\mathbf{c}'_j$ from $\mathbf{C}_l$ to obtain unique cluster mappings. In line 8, we then map the sorted attribute values in $\mathbf{c}_j.\mathbf{v}$ to the corresponding values in cluster $\mathbf{c}'_j.\mathbf{v}$, such that each attribute value from data set $\mathbf{D}^s$ has a unique mapping to a value in the lookup table $\mathbf{L}$.

4. Generating an Anonymised Data Set

Once the attribute value mapping has been completed, DOYEN generates the anonymised graph data set $\mathbf{D}^a$ for the sensitive input graph data set $\mathbf{D}^s$ in the following manner.

For each record from the input data set $r_i \in \mathbf{D}^s$, we create a synthetic record $r'_i \in \mathbf{D}^a$, and for each sensitive attribute value in r_i, we replace the original attribute value with the corresponding mapped attribute value from $\mathbf{M}$.

Given many data sets have dates associated with their records (such as dates of birth or dates of hospital admission), DOYEN also provides an anonymisation approach for date values while maintaining the temporal distances across connected records. Prior to date anonymisation, the records in $\mathbf{D}^s$ are grouped such that related records are contained in a single group. These groupings reflect records that represent a related group of entities, such as the siblings of the same family. Subsequent to grouping records, DOYEN sorts the date values associated with the records in a group. The tool allows the user to define a minimum (d_{min}) and a maximum (d_{max}) date within which they want the earliest date value from a specific group to appear. Thus, for the earliest date d_1 from a record group, we create a corresponding date d'_1 where $d_{min} \leq d'_1 \leq d_{max}$. Then, the remaining date values in the record group are shifted by $d'_1 - d_1$ days and each newly generated date value, excluding the earliest date d'_1, is randomly shifted by $\pm\Delta d$ days such that any temporal constraints across the generated date values are maintained. For example, if $\mathbf{D}^s$ contains birth records of sibling groups, then the temporal constraints of the birth dates would reflect that it is not possible for two births by the same mother to be less than nine months apart unless they are twins [8]. The anonymised date values are then used in the synthetic records $r'_i \in \mathbf{D}^a$.

Table 1
Number of unique attribute values

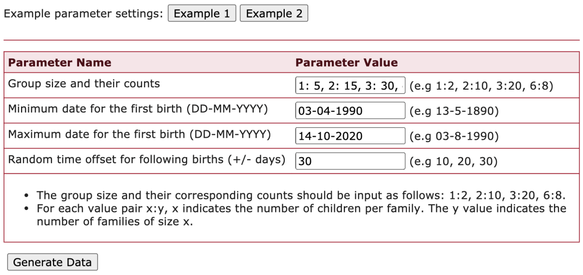
Attribute	First name		Surname	Address
	F	M		
Lookup tables	33,334	19,151	51,350	2,255
Data set 1	274	231	152	124
Data set 2	375	333	266	219

5. Demonstration

The initial implementation of the DOYEN tool demonstrates sensitive graph data anonymisation using two input data sets which we generated based on two real-world historical birth data sets. These data sets contain name and address variations to help demonstrate the capability of DOYEN to anonymise a graph while maintaining its similarity structure. The example birth data sets contain several twin births as well as missing values for the last names of fathers and children, as seen in the original birth data sets. Lookup tables containing values for the sensitive attributes first name, surname, and address, were generated using a publicly available US voter database (see: https://dl.ncsbe.gov) as well as Australian addresses. Table 1 summarises the number of unique values in each sensitive attribute in the lookup tables and the two example data sets. If attribute values contained more than one token (such as having two names) they were separated into individual tokens as a preprocessing step.

As shown in Table 1, the number of values available for each attribute is significantly larger in the lookup tables compared to the input graph data set. Having more attribute values in the lookup tables will help anonymise the sensitive input data set in a manner that better preserves its graph similarity structure.

Fig. 2 shows the input screen of the DOYEN web tool. The two buttons *Example 1* and *Example 2* will load one of the input graph data sets and suitable pa-



Example parameter settings:

Parameter Name	Parameter Value
Group size and their counts	1: 5, 2: 15, 3: 30, (e.g 1:2, 2:10, 3:20, 6:8)
Minimum date for the first birth (DD-MM-YYYY)	03-04-1990 (e.g 13-5-1890)
Maximum date for the first birth (DD-MM-YYYY)	14-10-2020 (e.g 03-8-1990)
Random time offset for following births (+/- days)	30 (e.g 10, 20, 30)

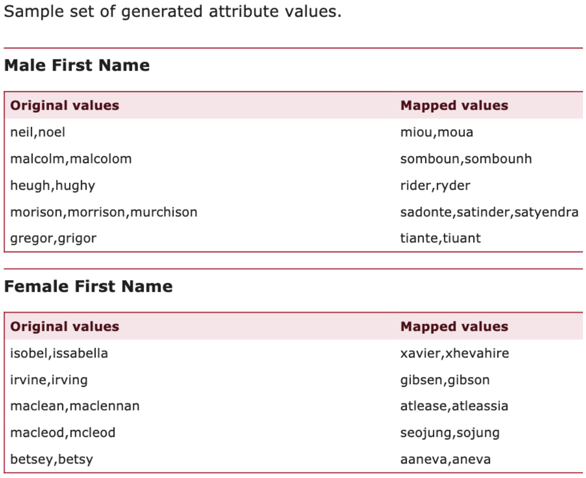
- The group size and their corresponding counts should be input as follows: 1:2, 2:10, 3:20, 6:8.
- For each value pair x:y, x indicates the number of children per family. The y value indicates the number of families of size x.

Figure 2: Input screen of the DOYEN tool.

rameter values. A sample of the input data set can be viewed by clicking on the *View Input Data Sample* button, whereas the full data set can be downloaded as well. The four parameters to be specified are:

- *Group size and their counts*: This parameter allows control of the number of families of a given size that are to be generated. When a sensitive graph data set is loaded, this field is automatically populated with the family size distribution of the loaded input data set. The user can change the values as desired. However, the current implementation restricts the user to specifying only up to a maximum number of families as appearing in the input data set, for a given family size. That is, if there are ten families (clusters) of size five in the input data set, then currently the user can only generate a maximum of ten families of size five.
- *Minimum/Maximum dates for the first birth*: For each family in the anonymised data set, the first birth record in the family is expected to have a birth date within (including) the given minimum and maximum date range (d_{min} and d_{max}), where $d_{min} < d_{max}$.
- *Random time offset*: This is the $\pm\Delta d$ time range which is used to further shift (randomly perturb) the dates of birth in each anonymised synthetic family (except the first date) as we described in Section 4.

The user has the flexibility of editing the values with which the parameter fields are automatically populated after an example data set has been loaded. When the *Generate Data* button is clicked, the anonymisation and data generation process is executed in the back end. The tool internally calculates the string similarity of attribute values by first applying a blocking technique [9] (with phonetic encoding for names and Locality Sensitive Hashing for addresses) and then applying a pairwise string similarity calculation method (we used Jaro-Winkler for names and the Jaccard q-gram based approach for addresses) [6]. Subsequently, the attribute value clusters are identified with the connected component clustering approach with a similarity threshold of 0.8 [9]. Next, all attribute value clusters from the input graph are mapped to clusters from the lookup table using the Euclidean distance vector similarity measure, as we described in Algorithm 1. Since the vector of similarities between attribute value pairs in clusters ($\mathbf{c}_j.\mathbf{s}$) is more important than the vector of attribute value lengths ($\mathbf{c}_j.\mathbf{l}$), we assign a relatively higher weight $w(w > 0.5)$ to $\mathbf{c}_j.\mathbf{s}$ and a weight of $1 - w$ to $\mathbf{c}_j.\mathbf{l}$ when calculating the overall cluster similarity.


Sample set of generated attribute values.

Male First Name

Original values	Mapped values
neil,noel	miou,moua
malcolm,malcolom	somboun,sombounh
heugh,hughy	rider,ryder
morison,morrison,murchison	sadonte,satinder,satyendra
gregor,grigor	tiante,tiuant

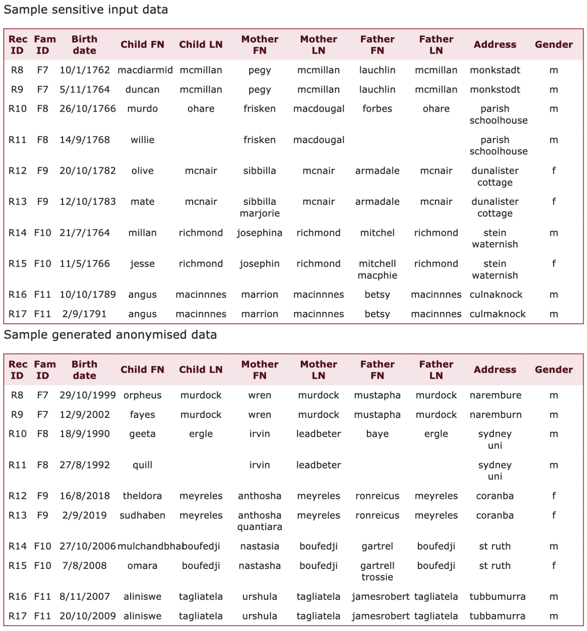
Female First Name

Original values	Mapped values
isobel,issabella	xavier,xhevahire
irvine,irving	gibsen,gibson
maclean,maclennan	atlease,atleassia
macleod,mcleod	seojung,sojung
betsey,betsy	aaneva,aneva

Figure 3: Sample of attribute value mappings as generated by DOYEN.

After the attribute value mapping is completed, the DOYEN tool generates the anonymised graph data set by replacing the attribute values of each record in the input data set with the corresponding mapped attribute value, and by shifting the dates of birth as described in Section 4. Once the anonymised data is generated, the user can view a sample of the attribute value mappings as shown in Fig. 3. Furthermore, the user can view a sample of the anonymised, synthetic data set created by DOYEN, or download the full data set. Fig. 4 shows a sample of the sensitive input data set and the corresponding anonymised records. Notice how, for example, the address values 'monkstadt' and 'monkstodt' (in R8 and R9) with a high pairwise string similarity have been replaced with values 'narembure' and 'narembure' which have a similar pairwise similarity.

To illustrate the quality of the anonymised graphs generated by DOYEN, Fig. 5 shows the distribution of the pairwise similarity for record pairs from each example data set. For each record pair, the similarity is



Sample sensitive input data

Rec ID	Fam ID	Birth date	Child FN	Child LN	Mother FN	Mother LN	Father FN	Father LN	Address	Gender
R8	F7	10/1/1762	macdiarmid	mcmillan	pegy	mcmillan	lauchlin	mcmillan	monkstadt	m
R9	F7	5/11/1764	duncan	mcmillan	pegy	mcmillan	lauchlin	mcmillan	monkstodt	m
R10	F8	26/10/1766	murdo	ohare	frisken	macdougal	forbes	ohare	parish schoolhouse	m
R11	F8	14/9/1768	willie		frisken	macdougal			parish schoolhouse	m
R12	F9	20/10/1782	olive	mcnair	sibbilla	mcnair	armadale	mcnair	dunalister cottage	f
R13	F9	12/10/1783	mate	mcnair	sibbilla marjorie	mcnair	armadale	mcnair	dunalister cottage	f
R14	F10	21/7/1764	millan	richmond	josephina	richmond	mitchel	richmond	stein waternish	m
R15	F10	11/5/1766	jesse	richmond	josephin	richmond	mitchell macphie	richmond	stein waternish	f
R16	F11	10/10/1789	angus	macinnnes	marrion	macinnnes	betsy	macinnnes	culnaknock	m
R17	F11	2/9/1791	angus	macinnnes	marrion	macinnnes	betsy	macinnnes	culmaknock	m

Sample generated anonymised data

Rec ID	Fam ID	Birth date	Child FN	Child LN	Mother FN	Mother LN	Father FN	Father LN	Address	Gender
R8	F7	29/10/1999	orpheus	murdock	wren	murdock	mustapha	murdock	narembure	m
R9	F7	12/9/2002	fayes	murdock	wren	murdock	mustapha	murdock	naremburn	m
R10	F8	18/9/1990	geeta	ergle	irvin	leadbeter	baye	ergle	sydney uni	m
R11	F8	27/8/1992	quill		irvin	leadbeter			sydney uni	m
R12	F9	16/8/2018	theldora	meyreles	anthosha	meyreles	ronreicus	meyreles	coranba	f
R13	F9	2/9/2019	sudhaben	meyreles	anthosha quantiara	meyreles	ronreicus	meyreles	coranba	f
R14	F10	27/10/2006	mulchandbha	boufedji	nastasia	boufedji	gartrel	boufedji	st ruth	m
R15	F10	7/8/2008	omara	boufedji	nastasha	boufedji	gartrell trossie	boufedji	st ruth	f
R16	F11	8/11/2007	aliniswe	tagliatela	urshula	tagliatela	jamesrobert	tagliatela	tubbumurra	m
R17	F11	20/10/2009	aliniswe	tagliatela	urshula	tagliatela	jamesrobert	tagliatela	tubbamurra	m

Figure 4: Sample of the sensitive input data set and the anonymised data set as generated by DOYEN.

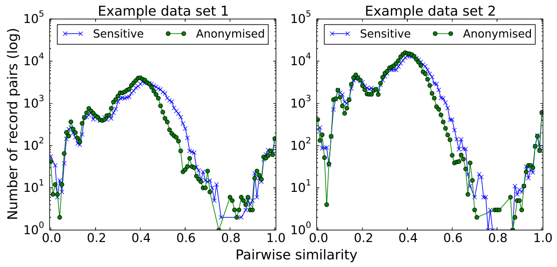


Figure 5: Comparison of pairwise record similarities of the sensitive input and the generated anonymised data sets.

calculated using the Jaro-Winkler similarity measure on names and the Jaccard q-gram similarity measure on addresses [6], followed by an averaging of these values. As can be seen from this figure, the similarity distribution of both the sensitive input data set and the generated anonymised data set are highly similar for each of the two example data sets. This shows the capability of DOYEN to anonymise sensitive graph data, while maintaining its structure, as reflected by these pairwise similarities.

6. Conclusion and Future Work

In this paper, we have presented the initial implementation of our web tool DOYEN which has the capability of anonymising sensitive graph data sets while preserving their similarity structure. As future work we intend to further improve DOYEN such that it is capable of maintaining geographic structures in the graph data (such as maintaining similar distances between addresses in input records and the corresponding generated anonymised output records). Furthermore, we plan to support generating synthetic data sets of different cluster size distributions without restricting the user to a maximum distribution as limited by the input data set. Such flexibility will allow users to generate larger or smaller data sets as suited for their research requirements.

Acknowledgments

This research was partially funded by the Australian Research Council under grant DP160101934.

References

[1] T. Feder, S. U. Nabar, E. Terzi, Anonymizing graphs, arXiv Preprint (2008). URL: http://arxiv.org/abs/0810.5578.

[2] L.-E. Wang, X. Li, A graph-based multifold model for anonymizing data with attributes of multiple types, CaS 72 (2018) 122–135.

[3] S. Das, O. Egecioglu, A. Abbadi, Anónimos: An LP-based approach for anonymizing weighted social network graphs, IEEE TKDE 24 (2012) 590–604.

[4] B. Zhou, J. Pei, W. Luk, A brief survey on anonymization techniques for privacy preserving publishing of social network data, SIGKDD Explor. Newsl. 10 (2008) 12–22.

[5] G. Bloothooft, P. Christen, K. Mandemakers, M. Schraagen, Population Reconstruction, Springer, Cham, 2015.

[6] G. Navarro, A guided tour to approximate string matching, ACM Computing Surveys 33 (2001) 31–88.

[7] R. Xiang, J. Neville, M. Rogati, Modeling relationship strength in online social networks, in: WWW, ACM, Raleigh, NC, 2010, pp. 981–990.

[8] C. Nanayakkara, P. Christen, T. Ranbaduge, Robust temporal graph clustering for group record linkage, in: PAKDD, Springer, Macau, 2019.

[9] P. Christen, Data Matching – Concepts and techniques for record linkage, entity resolution, and duplicate detection, Springer, Heidelberg, 2012.