Report on De-anonymizing Social Networks by Arvind Narayanan and Vitaly Shmatikov

Bhandaru Rohith - EE13B016, Yadugiri Saikumar - EE14B067

Indian Institute of Technology Madras

Abstract

This paper is a report on the paper De-anonymizing Social Networks published on 2009 for the 30th IEEE Symposium on Security and Privacy. The authors of the paper are Arvind Narayanan and Vitaly Shmatikov. The event took place at University of Texas, Austin. Operators of Social Networks are sharing their sensitive information to many vendors may it be for trageted advertisement or for research. This data is typically anonymized by removing the sensitive information about people like names addresses etc. But in this paper, the authors argue that this is not enough. They present an algorithm which analyzes uses from Twitter and Flickr and de-anonymize some of the users who are present in the anonymized Twitter graph. This was done without any use of "sybil" nodes and present a case where sybil nodes are easy to trace. The error rate is surprisingly low. It is 12%.

Keywords: Privacy, De-anonymization, Learning Algorithms

1. Introduction

Many people think that privacy in an online world is protected by anonymization. But does it really? Recent proliferation in the amount of social networking websites like Facebook, Twitter etc. have not only attracted many people but also computer scientists. In this paper the authors present a framework to analyse privacy and anonymization, also develop a new re-identification algorithm and test it on both Twitter and Flickr. The result is that users in the Flickr network can be re-identified in the Twitter graph too with an error rate of 12%. This algorithm works even when the overlap between the target network and the adversary's auxiliary information is small.

Many of the social network website operators sell information about their clients to advertisement companies usually removing their names, addresses. This is what they think to be called anonymization. But in this paper we will show that even anonymization is insufficient for perfectly hiding the privacy. We will define a security breach and form a re-identification algorithm which identifies the nodes of a anonymized graph with non-negligible probability.

2. Motivation

Currently, most of the data-releases happen for the purpose of –

1. Academic and government data-mining.
2. Advertising
3. Third party applications
4. Aggregation.

The data released for data-mining in government and academic sectors can be vulnerable to leak. This happened once in an anonymous Midwestern high school as part of the Add Health project, a detailed survey on adolescent health. With emergence of digital advertising, many big companies including Google and Facebook have released data for advertisers but this might be vulnerable to a leak. Many third party applications use APIs from social media companies and misuse the information they are given. As the data given to them will be in un-anonymized form, we have a higher risk of security breach with respect to these fields. Many online data projects such as OpenID, Jaiku etc release data on a regular basis. Apart from these, many friend-to-friend networking sites, a peer-to-peer file sharing organisations might also have their traffic typically not anonymized. These all lead to acquirement of auxiliary data about the network.

Many related works have been done in this are including some practical attacks. There are O(log N) Sybil attacks that were proposed too. But they can be identified as we would have a subset of nodes with no out edges. The cut-based attack creates 7-node subgraphs containing a Hamiltonian path. But this too only as long as a small percentage of users link back to the Sybil nodes. Even for the definitions of privacy, we have different models such as k-anonymity etc. But all these existing models fail to capture self-reinforcing, feedback based attacks. In this paper, authors address this issue.

3. Problem

Given a large amount of data, can we re-identify the nodes in the network graph with the help of some other auxiliary information or not?

4. Solution

Before we dive into the solution, let us define some quantities useful for the better understanding of the paper.

4.1. Identifying and Non-identifying Attributes

The EU law on personal information states the following about identifying attributes -

> ”Any information relating to an [..] natural person who can be identied, directly or indirectly, in particular by reference [..] to one or more factors specic to his physical, physiological, mental, economic, cultural or social identity”

The same has been adapted by the Indian Law. Anything which doesn't follow the above rule is called non-identifying attribute. In this paper, we are going to discuss re-identification and de-identification algorithms.

4.2. Social Network

We define a social network as a directed graph G = (V,E). The vertex set corresponds to people or entities in the network. The edge set corresponds to the relationships of the entities. This may need different in different networks. For example in Twitter, each edge is an existence of following the other node. In Facebook, it means the "friend" entity. A set of attributes corresponding to the nodes and edges are present. We denote these by $\mathcal{X}$ and $\mathcal{Y}$ respectively. Note that we define all edge attributes over V^2 and not on E. This is mainly because we do not know if there is a relationship edge or not based on the given information.

If $(u, v) \notin E$, then Y[u,v] = $\perp$, $\forall\ Y \in \mathcal{Y}$.

4.3. Data Sanitization and Release Process

We don't get the entire network when asked for. We only get a subgraph which is ananymized too. So, we will have $V_{san} \subset V$, $\mathcal{X}_{san} \subset \mathcal{X}$ and $\mathcal{Y}_{san} \subset \mathcal{Y}$. Also the data we are receiving is inserted with some error.So, the final data released is -

$$(V_{san},\ E_{san},\ \{X(v), \forall v \in V_{san}, X \in \mathcal{X}_{san}\}\ ,\ \{Y(e), \forall e \in E_{san}, Y \in \mathcal{Y}_{san}\})$$

So, now the question is that can we use this anonymized data to get sensitive information about an individual. This information can be obtained from many online data mining websites.

4.4. Attacker Model

The attacker is assumed to be not only have the anonymized data but also auxiliary information about the network, S_{aux}. There can be a partial or no overlap between these two data. The existence of such information is non-trivial. It can be obtained in many ways. For instance the attacker himself might be a part of the target network and he can obtain the information about his neighbours in the network or he can screen-scrape data. As we have the auxiliary information, we can know some information about nodes and their attributes before hand. The can be thought of as prior probabilities. Aux_X and Aux_Y can be thought of as probability distributions over V_{aux} and E_{aux}.

Aux[X,v] :Attacker's prior probability distribution of the value of the attribute X of node v.
Aux[Y,e] :Attacker's prior probability distribution of the value of the attribute Y of edge e.

It is also assumed that the attacker has information about very small number of nodes in the target network S. This can be due to his presence or in many ways. Note that the attacker's motive can be anything from a small prank to global surveillance. Even though the attacker has possession of large S_{aux}, de-anonymization of S is non-trivial.

4.5. Privacy Breach

Before we look at the algorithm, we need to know when the algorithm succeeds. So, we define the notion of privacy breach. Many people have different ides of a privacy breach. But throughout the paper we are going to stick with this definition of privacy breach. We define a privacy policy function, PP defined as -

$$\text{PP} : \mathcal{X} \cup \mathcal{Y} X E \rightarrow \{pub, priv\}$$

As we can see this is a boolean function which defines every attribute to be either private or public. This is in accordance with identifying and non-identifying attributes definition.

4.6. Ground Truth

The Ground Truth as the name suggests is a mapping μ_G from the nodes of V_{aux} to V_{san}. $\mu_G =\perp$ if there is no node in V_{san} corresponding to v in V_{aux}. The main aim of our re-identification algorithm is to find this ground truth i.e, on input S_{san} and S_{aux} the re-identification algorithm outputs a probabilistic mapping $\tilde{\mu}$ defines as -
$\tilde{\mu} : V_{san}X(V_{aux} \cup \{\perp\}) \rightarrow [0, 1]$ where $\tilde{\mu}(v_{aux}, v_{san})$ is the probability that v_{aux} maps to v_{san}.

4.7. Mapping Adversary

We define a mapping adversary corresponding to the probabilistic mapping $\tilde{\mu}$ as the probability distribution -

$$\mathrm{Adv}[\mathrm{X},\mathrm{v}_{aux},\mathrm{x}] = \frac{\sum_{v \in V_{san}, X[v]=x} \tilde{\mu}(v_{aux}, v)}{\sum_{v \in V_{san}, X[v]\neq\perp} \tilde{\mu}(v_{aux}, v)}$$

$$\mathrm{Adv}[\mathrm{Y},\mathrm{u}_{aux},\mathrm{v}_{aux},\mathrm{y}] = \frac{\sum_{u,v \in V_{san}, Y[u,v]=y} \tilde{\mu}(u_{aux}, u)\tilde{\mu}(v_{aux}, v)}{\sum_{u,v \in V_{san}, Y[u,v]\neq\perp} \tilde{\mu}(u_{aux}, u)\tilde{\mu}(v_{aux}, v)}$$

4.8. Privacy Breach

For nodes u_{aux}, $\mathrm{v}_{aux} \in \mathrm{V}_{aux}$, let $\mu_G(u_{aux}) = u_{san}$ and $\mu_G(v_{aux}) = v_{san}$. We say that the privacy of v_{san} is breached w.r.to the mapping adversary Adv and privacy parameter δ if -

1. For some attribute X such that PP[X] = priv, x = X[v_{aux}], Adv[X,v_{aux},x]-Aux[X,v_{aux},x] $> \delta$ OR

2. For some attribute Y such that PP[Y] = priv, y = Y[u_{aux},v_{aux}], Adv[Y,u_{aux},v_{aux},y]-Aux[Y,u_{aux},v_{aux},y] $> \delta$

4.9. Success of De-anonymization

Let V_{mapped} = v $\in V_{aux} : \mu_G(v) \neq\perp$. The success rate of de-anonymization algorithm giving a probabilistic mapping $\tilde{\mu}$ as output with respect to a centrality measure ν is the probability that μ sampled from $\tilde{\mu}$ maps a node v to μ_G(v) weighted with ν(v) as follows:

$$\text{Success rate} = \frac{\sum_{v \in V_{mapped}} Pr[\mu(v)=\mu_G(v)].\nu(v)}{\sum_{v \in V_{mapped}} \nu(v)}$$

But this is only a lower bound.

5. Re-identification Algorithm

Using the definitions in the previous section, we define the re-identification algorithm as follows -

1. Identify a set of nodes present in both S_{san} and S_{aux} and map them to one another. We call these seed nodes.

2. This seed map is propagated to other nodes in the network through an iterative process based only on the topology of the network.

But the seed identification is not straight-forward. What one can do is find a k-clique in the auxiliary network. As this is an auxiliary network, we assume that the attacker knows the degree and neighbours of the nodes in the clique. We search for the same in the sanitized network given to us within a factor of $(1 \pm \epsilon)$. Then we maintain a list of all the mapped nodes. In each iteration, an unmapped node u in S_{aux} is selected and scores are calculated for every unmapped in S_{san}. Here the score(u,v) is defined as the number of neighbours of u mapped to the neighbours of v. If the score is above a threshold, then u is mapped to v. We call the strength of the score which is a heuristic as eccentricity. We also use several other heuristics like edge directionality, node degree normalization and reverse mapping in the algorithm. The following is the pseudo-code for the algorithm.

```
function propagationStep(lgraph, rgraph, mapping)

for lnode in lgraph.nodes:

        scores[lnode] = matchScores(lgraph, rgraph, mapping, lnode)

        if eccentricity(scores[lnode]) < theta: continue

        rnode = (pick node from right.nodes where

         scores[lnode][node] = max(scores[lnode]))

scores[rnode] = matchScores(rgraph, lgraph, invert(mapping), rnode)

        if eccentricity(scores[rnode]) < theta: continue

        reverse_match = (pick node from lgraph.nodes where

         scores[rnode][node] = max(scores[rnode]))

        if reverse_match != lnode:

         continue

mapping[lnode] = rnode

function matchScores(lgraph, rgraph, mapping, lnode)

        initialize scores = [0 for rnode in rgraph.nodes]

        for (lnbr, lnode) in lgraph.edges:

         if lnbr not in mapping: continue
```

```
            rnbr = mapping[lnbr]

            for (rnbr, rnode) in rgraph.edges:

             if rnode in mapping.image: continue

                 scores[rnode] += 1 / rnode.in_degree ^ 0.5
for (lnode, lnbr) in lgraph.edges:

            if lnbr not in mapping: continue

            rnbr = mapping[lnbr]

            for (rnode, rnbr) in rgraph.edges:

             if rnode in mapping.image: continue

                 scores[rnode] += 1 / rnode.out_degree ^ 0.5
return scores

    function eccentricity(items)

        return (max(items) - max2(items)) / std_dev(items)

until convergence do:

        propagationStep(lgraph, rgraph, seed_mapping)
```

6. Results

Based on the experiments conducted using the above algorithm and networks from Twitter and Flickr, the authors have published the following results -

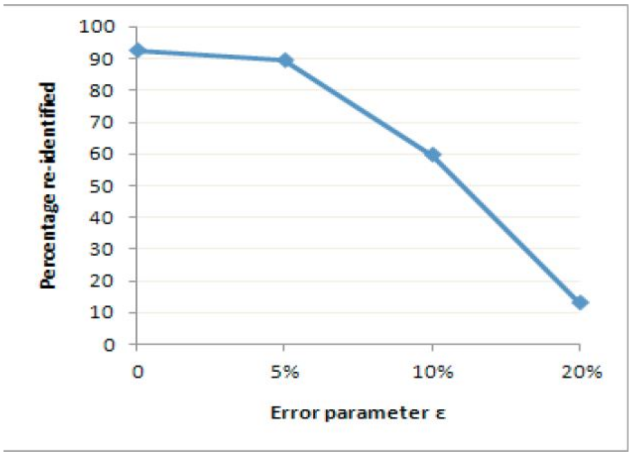

Figure 1: Re-identication rate decreases with noise parameter

The figure illustrates the effect of noise used in the anonymized data given to us. We can see that as the noise.

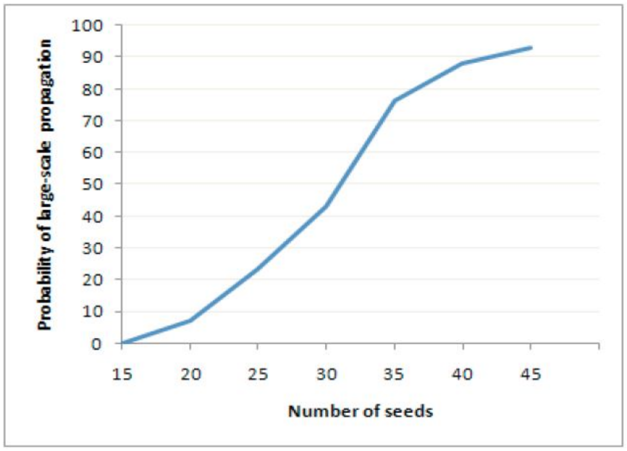

Figure 2: Phase transition in scale of re-identication Vs. Number of seeds

The figure illustrates the effect of number of seeds on the phase-transition. If we can identify many seeds at the start of the algorithm, we can have higher transition.

7. Conclusion

It is interesting to note that even though the paper deals with the importance of privacy, it doesn't propose a mechanism to actually find a better way for anonymization. Based on the data of the experiments we can see that anonymization doesn't actually mean privacy. We can also use the information acquired in the current run to know more about the person in the next run. It is better to have a query-based release of data rather than release and forget approach.

8. References

[1] Narayanan, Arvind, and Vitaly Shmatikov. *"De-anonymizing social networks."* Security and Privacy, 2009 30th IEEE Symposium on IEEE, 2009.

[2] Narayanan, Arvind, and Vitaly Shmatikov. *"Myths and fallacies of personally identiable information."* Communications of the ACM 53.6 (2010): 24-26.

[3] Directive 95/46/EC of European Parliament

[4] Ohm, P. *Broken promises of privacy: Responding to the surprising failure of anonymization.* 57 UCLA Law Review 57, 2010