On the Protection of Social Network-Extracted Categorical Microdata

Jordi Marés[1] and Vicenç Torra[2]

[1] Artificial Intelligence Research Institute (IIIA),
Spanish Council of Scientific Research (CSIC),
Universitat Autònoma de Barcelona (UAB), Spain
jmares@iiia.csic.es
[2] Artificial Intelligence Research Institute (IIIA),
Spanish Council of Scientific Research (CSIC), Spain
vtorra@iiia.csic.es

Abstract. Social networks have become an essential part of the people's communication system. They allow the users to express and share all the things they like with all the people they are connected with. However, this shared information can be dangerous for their privacy issues. In addition, there is some information that is not explicitly given but is implicit in the text of the posts that the user shares. For that reason, the information of each user needs to be protected.

In this paper we present how implicit information can be extracted from the shared posts and how can we build a microdata dataset from a social network graph. Furthermore, we protect this dataset in order to make the users data more private.

1 Introduction

With the continuous growing amount of public available data, individual privacy has become a very important issue to deal with because several agencies are collecting a huge amount of data from people daily. This data is very valuable for the knowledge of our society status but it is also dangerous in terms of privacy. Data privacy field tries to protect all the public data sources in order to allow the data extraction but taking into account the individuals privacy. Until a few years ago, the major part of the data was collected via surveys. However, nowadays there is a new place to take data much more easy and much less controlled: the online social networks.

Social networks have become a very important part of the people's communication system and, as most sociologists agree, this online social interaction will not fade away [18]. People use these networks to express all their feelings, emotions or simply to meet people who have the same hobbies or interests. It can be seen that all this information is sensible and is related to a single user profile. Therefore it can be dangerous to collect this kind of data and publish it without protection. An example of the need to protect social networks can be found in [19] where it says that epidemiology researchers use social networks to study the social network structure and epidemic phase in sexually transmitted disease. In addition it should be noticed that not all the information is explicitly given by the user profile. There is some information that is implicitly hidden into the posts the user shares in his profile such as the main topics of interest of the user.

J. Nin and D. Villatoro (Eds.): CitiSens 2012, LNAI 7685, pp. 33–42, 2013.

Although there have been several approaches to protect the user anonymity modifying the social graph structure adding or removing edges [12][23], there are less approaches to deal with the privacy in the semantic data included in the graph nodes [4]. The most well known model to protect social graphs is k-anonymity which is a very popular model for microdata datasets protection [14] and it has been adapted to graphs [9] and relies on the property that every node will be indistinguishable with at least $(k-1)$ nodes.

In this paper we present a way to protect a real online social network-extracted microdata dataset with explicit and implicit information about Twitter users using a k-anonymity protection method. Several approaches have been developed to protect microdata datasets [2][17][20] in order to achieve enough protection to prevent attacks to the confidential information about individuals from the disseminated data.

Regarding the data in the microdata datasets, there exist two types: categorical and continuous. In our case, we focus on categorical data. The problem of categorical data over continuous data is that there are less actions to perform in the protection process because arithmetic operations are not allowed here, so the only actions allowed with categorical data are the exchange of categories by others that already exist, suppression of category, and generalizations of some categories into new ones. This lack of possible operations makes the protection a difficult task.

Protection methods are typically evaluated using two measures: information loss and disclosure risk. Information loss [17] checks the quantity of data that has been harmed during the protection process and therefore is no longer useful. Disclosure risk [6][21][22] measures the quantity of original data that can be discovered through the protected data.

The remaining of this paper is structured as follows. In Section 2 we explain the methodology followed to go from a real social network like Twitter to obtaining a microdata dataset with explicit and implicit information about users. Section 3 contains the description of the protection method used in this work to protect the microdata dataset: the microaggregation. In Section 4 we present the measures used to evaluate the quality of the protection. Section 5 shows the results of our experiments comparing privacy and utility in the original microdata dataset and the generated protections. In Section 6 we make some concluding remarks. Finally, in Section 7 we describe our next steps to do as a future work.

2 Social Network-Extracted Microdata Generation

In this section we describe the methodology we used in order to extract a microdata dataset from a real online social network like Twitter.

2.1 Crawling Algorithm

The first step to take is to build a crawler in order to get information about connected users in the social network. Algorithm 1 shows the steps followed by our crawler.

Algorithm 1. Twitter Profiles Crawling Algorithm

Input: uID Initial user id, $numUsers$ Maximum number of user to crawl, $numTweets$ Number of tweets to get from each user.
Output: Y List of public available data for each user.
id $\Leftarrow$ uID
actualUser $\Leftarrow$ $getDataFromUser$(id,numTweets)
unvisited$\Leftarrow$ $getFollowingUsers$(actualUser)
visited $\Leftarrow$ [id]
$Y \Leftarrow$ [actualUser]
while ($|unvisited| > 0$) and ($|visited| < numUsers$) **do**
  id $\Leftarrow$ $getRandomId$(unvisited)
  actualUser $\Leftarrow$ $getDataFromUser$(id,numTweets)
  unvisited.$remove$(id)
  newRemaining $\Leftarrow$ $getFollowingUsers$(actualUser)
  unvisited.add(newRemaining)
  visited.add(id)
  Y.add(actualUser)
end while
return Y

The algorithm is started with a given initial user id as the starting node in the social network, a maximum number of users we want to get information from, and a number of tweets we want to get from each user. Then, we use the Twitter API [3] to get user data such as location, hashtags, urls, following users, and tweets posted by the user. Three lists are used: *unvisited* contains the ids of the not yet crawled users connected to the already crawled ones, *visited* contains the ids of the already crawled users, and *Y* contains the data structures containing all the information about each crawled user.

This is executed in a loop until we reach the maximum number of users we wanted to crawl or until we have no more users in the *unvisited* list.

After this step we have a collection of structures containing information about each user.

2.2 User Profiles Generation

The second step to do is to use the data structures collected by the crawler in order to get a profile for each user containing his location, his connected users and, his three most relevant topics of interest. In order to do this it should be noticed that information is not always explicitly given in the social networks. That is, using the Twitter API we can get the location but it is not possible to get the topics that a user is interested about because they are not specified nor described anywhere. However, these topics can be extracted using natural language processing techniques on the text of the tweets shared by the user.

In order to process the information contained in the tweets we used Web services provided by OpenCalais [15], which allow for the extraction of entities such as people, organizations or events and moreover assign topics to a piece of text. In this work we only used the topics categorization capacities of OpenCalais. The 18 possible topic output

values are: Business_Finance, Disaster_Accident, Education, Entertainment_Culture, Environment, Health_Medical_Pharma, Hospitality_Recreation, Human Interest, Labor, Law_Crime, Politics, Religion_Belief, Social Issues, Sports, Technology_Internet, Weather, War_Conflict and, Other.

Our first approach was to apply directly the OpenCalais Web services to the tweets text. However, as tweets are very short pieces of text (maximum of 140 characters) it was very difficult to extract topics and we got a very high percentage of users without any topic of interest found. Then, as a second approach, we used the urls within the tweets texts to enhance their semantics following the approach described in [1].

In this work, we do not use the hashtags because most of the times they are written in a useless form such as $\#ToMyFutureKids$. This forms do not provide any information to us and therefore we decided to not use hashtags but use the web pages shared in the tweets, which are much more rich semantically.

To do this, we executed two times the OpenCalais Web service to check the topics found in the tweet text and also in the text of the website shared inside the tweet. Then, the topics found in both executions were merged. At the end of processing all the tweets from a given user, the three most frequent topics were the ones taken as a result. By doing this we obtained a higher number of topics per user. The final topics also kept the level of interest for each topic because we took the most frequent one as the main topic of interest for the user, the second most frequent is the second main topic of interest, and the same happens for the third.

At the end of this profiles generation step we have a set of user profiles containing the location of a user, the users who is connected with, and the sorted three major topics of interest. So, as a result we obtained profiles combining explicit information given by the Twitter API calls and implicit information extracted from the tweets shared by the user using natural language processing tools.

2.3 Graph Generation

As a third step, after generating the users profiles, we generated the social graph connecting all the users with the ones they are following in the real social network. Figure 1 shows the resulting graph representing the relations between users.

It can be seen that there are more density of edges in the center of the graph than in the borders. This is because when we crawled the social network we kept a list of remaining users to crawl which were connected users to already crawled users. This fact gives higher probabilities to the first crawled users to expand more their neighbors than to the last crawled users.

Then, as the initial user we crawled is in the center, all the users near to him had much more attempts to expand their neighbors than the users in the borders which are the newest ones.

2.4 Microdata Dataset Construction

Finally, once we have a social graph where each node has information about a single user and each user is connected to his real following users, it is possible to extract all this information from the nodes and generate a microdata dataset.

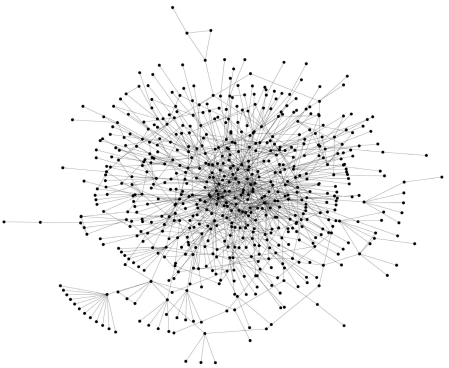
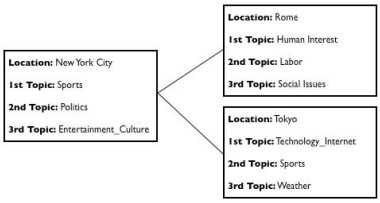


Fig. 1. Graph generated from the crawled users profiles

In order to do this we extracted the information of each node placing it in a single row of the dataset. Then, the resulting dataset file has one row per user and one column per attribute. In our case we used five attributes per user: the degree of the user node, the location of the user, the main topic of interest, the second main topic of interest, and the third main topic of interest. Figure 2 shows an example of microdata dataset construction from a graph with three user profiles.



Degree	Location	1st Topic	2nd Topic	3rd Topic
2	New York City	Sports	Politics	Entertainment_Culture
1	Rome	Human Interest	Labor	Social Issues
1	Tokyo	Technology_Internet	Sports	Weather

Fig. 2. Example of microdata construction

At the end of this step we have a real social network-extracted microdata dataset with either explicit and implicit information about the users. This kind of datasets would be very interesting for research purposes but they must be protected before publishing it.

3 The Microaggregation Protection Method

In this section we present the microaggregation protection method that is the one we have used to protect the microdata dataset in our approach.

In microaggregation [5][16][8], records are clustered into small aggregates or groups of size at least k. Then, instead of publishing an original variable V_i for a given record, the median of the values of V_i over the cluster to which the records belongs to is published.

To define the microaggregation procedure we need to define how to compute the distance between two categories when we create the clusters. This distance is defined in a different way when the variable is nominal than when it is ordinal because of the possibility of sorting the categories in the ordinal case, what is not possible in the nominal case.

For a nominal variable V the distance between two categories is defined as follows

$$d_{nominal}(c, c') = \begin{cases} 0 & \text{if } c = c' \\ 1 & \text{if } c \neq c' \end{cases} \tag{1}$$

and for an ordinal variable

$$d_{ordinal}(c, c') = \frac{|c'' : (c, c') \leq c'' \leq max(c, c')|}{|D(V)|} \tag{2}$$

where c is a category in the original dataset and c' is the category corresponding to c in the masked dataset, and $D(V)$ is the domain of variable V. Then, the ordinal distance will be the computed as the number of categories between c and c', divided by the total number of categories for the attribute V.

There exist several approaches for the microaggregation clustering. In this work we used the MDAV-generic described in [8] because it can work with any type of attribute, aggregation operator and distance. Algorithm 2 shows the algorithm of this method. Basically, MDAV create clusters of size k around the two most distant records in the dataset, leaving a final cluster with at least k records.

4 Protection Evaluation Measures

After protecting a microdata dataset it must be evaluated in order to assess the quality of the protection. In this paper we used the two main measures used in the microdata protection field: the information loss and the disclosure risk.

Information loss is known as the quantity of harm that is inflicted to the data by a given masking method. This measure is small when the analytic structure of the masked dataset is very similar to the structure of the original dataset. Then, the motivation for preserving the structure of the dataset is to ensure that the masked dataset will be analytically valid and interesting. In this work we used the *contingency table-based information loss* (CTBIL)[17], the *distance-based information loss* (DBIL)[17], and the *entropy-based information loss* (EBIL)[10].

Assessment of the quality of a protection method cannot be limited to information loss because disclosure risk has also to be measured. Disclosure risk measures the information can be obtained about the individuals from the protected data set. This measure is small when the masked dataset values are very different to the original values. In this work we used the *interval disclosure* (ID)[6], the *distance-based record linkage* (DBRL)[7], the *probabilistic record linkage* (PRL)[7], and the *rank swapping record linkage* (RSRL)[13].

Algorithm 2. MDAV-generic microaggregation algorithm

Input: X dataset, k level of anonymity.
Output: X' protected dataset.
while $(|X| > 3k)$ **do**
  Compute the average record $\bar{x}$ of all records in X. The average record is computed attribute-wise
  Consider the most distant record x_r to the average record $\bar{x}$ using appropriate distance
  Find the most distant record x_s from the record x_r
  Form two clusters c_r and c_s around x_r, and x_s where $|c_r| = k$ and $|c_s| = k$
  Take as a new dataset X the previous dataset X minus the records in c_r and c_s
end while
if there are between $3k - 1$ and $2k$ records in X **then**
  compute the average record $\bar{x}$ of the remaining records in X
  Find the most distant record x_r from $\bar{x}$
  Form a cluster c_r containing x_r and the $k - 1$ records closest to x_r
  Form another cluster containing the rest of the records
else
  Form a cluster with the remaining records
end if
return Y

The problem here is that both measures are inversely related so the higher information loss the lower disclosure risk, and the inverse. In order to perform a good protection there must be a minimized and balanced combination of both measures.

5 Experimental Results

In this section we present the results obtained for the protection of the Twitter-extracted microdata dataset using the microaggregation protection method.

The microdata dataset we used in our experiments contained 621 Twitter users profiles but only 324 users have an associated topic of interest. As all the users without associated topics of interest will be directly aggregated into a single cluster, we just focused on the protection of the ones that have some associated topics.

It should be noticed that our method is sensitive to the choice of the initial node in the sense of that each generated graph will be different. However, in order to just make an initial test of our method we used one single graph to run the experiments.

To protect this dataset, we generated 10 different microaggregation protections with different levels of k-anonymity. Then, we evaluated the original microdata dataset without protection and all the 10 protections in order to assess the lack of privacy in the original microdata dataset and to determine which would the best protection. The results are shown in Table 1.

As we explained in Section 4 it can be seen that information loss and disclosure risk measures are inversely related. Then, as we want to obtain good protections and the quality of these protections is described by two inverse related measures, the best protections will be the ones that have minimum values in both measures and that these values are balanced. Having this into account, it can be seen that the original microdata

Table 1. Results of the original and protected microdata evaluation

Dataset	Information Loss	Disclosure Risk
Original	0.00	99.54
Protected K=2	26.19	52.44
Protected K=3	32.13	43.58
Protected K=4	35.44	36.36
Protected K=5	41.50	31.63
Protected K=6	42.21	30.43
Protected K=7	46.05	28.02
Protected K=8	48.08	26.30
Protected K=9	49.88	24.26
Protected K=10	51.84	23.55

dataset obtained, as expected, a very bad results with a 99% of disclosure risk and a 0% of information loss. This is very bad because it means that almost all the users are exposed to the disclosure of their sensible information. However, if we take a look at the different protections results we can see that measures are more reduced and balanced.

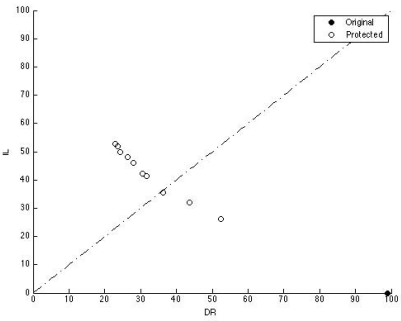


Fig. 3. Dispersion plot of the protected and original microdata evaluation results

Figure 3 shows the obtained results graphically. The dotted line represents the perfect balance of the measures so, the closest to the line and to the (0,0) point, the better protection. It can be seen that the original microdata is too far away from both. However, there is a protection that has the almost perfectly balanced values in both measures. Taking a look at Table 1 it can be seen that this is the case of the protection with k=4 (4-anonymity).

Comparing the results obtained in the original microdata and in the K=4 protection evaluations it can be seen that we have been able to decrease 63 points the risk of sensible information disclosure, but at the cost of increasing 35 points the analytically useful information. Then, we can be much more confident to publish this protected dataset than the original one in terms of individuals privacy.

Finally, it should be noticed that, as we are protecting a set of nodes attributes that include degree of each node, we are getting as a result a k-anonymous graph following the definition proposed by [11] that says that a graph is k-anonymous if every different

node degree appears at least in k nodes. Then, we can conclude saying that our protection approach could be used to perform this kind of k-anonymity protections.

6 Conclusions

In this paper we presented an approach to extract and protect microdata datasets from a real social network such as Twitter.

We have demonstrated that there is information that is not explicitly given in the social network user profile, but is implicit inside the posts the user shares. In order to get this kind of information we used the OpenCalais Web services to categorize the posts and extract the topics of interest from each user. In addition, in order to enrich the semantic content of the shared posts, we used the url's contained in the posts text.

We also have shown how to build a graph from the user extracted profiles, and how to convert it into a microdata dataset by taking the users profiles in the graph nodes.

Finally, we presented a way to protect this microdata dataset in order to be able to publish it for research purposes without violating the privacy of the contained users. We protected the dataset using the microaggregation method with different levels of k-anonymity. As a result we compared the evaluation of the privacy in the original dataset, and the protected ones. We demonstrated that the original dataset was violating the privacy of almost all the users, while using the microaggregation with 4-anonymity we obtained the best protection results reducing the risk of sensible data disclosure by 63 points but with the cost of increasing 35 points the loose of analytically useful information.

Then, we can conclude that microdata datasets can not only be extracted via surveys or statistical studies. They can also be extracted from the real social networks or graphs and, in this case, they may contain more information than the one explicitly described by the user in his social network profile. Then, they need also to be protected in order to publish them.

7 Future Work

In this work we only protected the dataset once it has been extracted from the social network. However, our main goal of the future work is to be able to protect the social graph information to get an already protected microdata dataset when it is extracted from the graph.

In addition, we also would like to consider the l-diversity rather than the k-anonymity since it has been proven that sometimes k-anonymity is not enough to protect a dataset.

Acknowledgments. This work has been done under the PhD in Computer Science program of the Universitat Autònoma de Barcelona (UAB). It is also partially supported by the Spanish MEC ARES-CONSOLIDER INGENIO 2010 CSD2007-00004.

References

1. Abel, F., Gao, Q., Houben, G.-J., Tao, K.: Semantic Enrichment of Twitter Posts for User Profile Construction on the Social Web. In: Antoniou, G., Grobelnik, M., Simperl, E., Parsia, B., Plexousakis, D., De Leenheer, P., Pan, J. (eds.) ESWC 2011, Part II. LNCS, vol. 6644, pp. 375–389. Springer, Heidelberg (2011)

2. Aggarwal, C., Yu, P.: Privacy-Preserving Data Mining: Models and Algorithms. Springer Publishing Company, Incorporated (2008)
3. Twitter API, https://dev.twitter.com
4. Campan, A., Truta, T.M.: Data and Structural *k*-Anonymity in Social Networks. In: Bonchi, F., Ferrari, E., Jiang, W., Malin, B. (eds.) PinKDD 2008. LNCS, vol. 5456, pp. 33–54. Springer, Heidelberg (2009)
5. Domingo-Ferrer, J., Mateo-Sanz, J.: Practical data-oriented microaggregation for statistical disclosure control. IEEE Trans. Knowl. Data Eng. 14(1), 189–201 (2002)
6. Domingo-Ferrer, J., Torra, V.: A quantitative comparison of disclosure control methods for microdata, pp. 111–133. Elsevier (2001)
7. Domingo-Ferrer, J., Torra, V.: Distance-based and probabilistic record linkage for re-identification of records with categorical variables. Butlletí de l'ÀCIA 28, 243–250 (2002)
8. Domingo-Ferrer, J., Torra, V.: Ordinal, continuous and heterogeneous *k*-anonymity through microaggregation. Data Min. Knowl. Discov. 11(2), 195–212 (2005)
9. Stokes, K., Torra, V.: Reidentification and k-anonymity: a model for disclosure risk in graphs. CoRR, abs/1112.1978 (2011)
10. Gouweleeuw, J., Kooiman, P., Willenborg, L.: Pram: A method for disclosure limitation of microdata. CBS research paper 9705 (1998)
11. Liu, K., Terzi, E.: Towards identity anonymization on graphs. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, pp. 93–106. ACM, New York (2008)
12. Nettleton, D.F., Sáez-Trumper, D., Torra, V.: A Comparison of Two Different Types of Online Social Network from a Data Privacy Perspective. In: Torra, V., Narakawa, Y., Yin, J., Long, J. (eds.) MDAI 2011. LNCS, vol. 6820, pp. 223–234. Springer, Heidelberg (2011)
13. Nin, J., Herranz, J., Torra, V.: Rethinking rank swapping to decrease disclosure risk. Data and Knowledge Engineering 64, 346–364 (2008)
14. Samarati, P., Sweeney, L.: Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report (1998)
15. OpenCalais Web Services, http://www.opencalais.com/calaisAPI
16. Torra, V.: Microaggregation for Categorical Variables: A Median Based Approach. In: Domingo-Ferrer, J., Torra, V. (eds.) PSD 2004. LNCS, vol. 3050, pp. 162–174. Springer, Heidelberg (2004)
17. Torra, V., Domingo-Ferrer, J.: Disclosure control methods and information loss for microdata, pp. 91–110. Elsevier (2001)
18. Tse, H.: An ethnography of social networks in cyberspace: The facebook phenomenon. The Hong Kong Anthropologist 2, 53–57 (2008)
19. Ward, H.: Prevention strategies for sexually transmitted infections: the importance of sexual network structure and epidemic phase. Sex Transm. Infect. (2007)
20. de Waal, T., Willenborg, L.: Elements of statistical disclosure control. Lecture Notes in Statistics. Springer (2001)
21. Winkler, W.: Re-identification methods for masked microdata (2004)
22. Yancey, W.E., Winkler, W.E., Creecy, R.H.: Disclosure Risk Assessment in Perturbative Microdata Protection. In: Domingo-Ferrer, J. (ed.) Inference Control in Statistical Databases. LNCS, vol. 2316, pp. 135–152. Springer, Heidelberg (2002)
23. Zheleva, E., Getoor, L.: Preserving the Privacy of Sensitive Relationships in Graph Data. In: Bonchi, F., Malin, B., Saygın, Y. (eds.) PInKDD 2007. LNCS, vol. 4890, pp. 153–171. Springer, Heidelberg (2008)