Knowledge-Based Systems 49 (2013) 134–144

Contents lists available at SciVerse ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

On the protection of social networks user's information

Jordi Marés [a,*], Vicenç Torra [b]

[a] Artificial Intelligence Research Institute (IIIA), Spanish Council of Scientific Research (CSIC), Universitat Autònoma de Barcelona (UAB), Bellaterra, Catalonia 08193, Spain
[b] Artificial Intelligence Research Institute (IIIA), Spanish Council of Scientific Research (CSIC), Bellaterra, Catalonia 08193, Spain

ARTICLE INFO

Article history:
Received 19 March 2013
Received in revised form 6 May 2013
Accepted 7 May 2013
Available online 30 May 2013

Keywords:
Social networks
Data privacy
k-Anonymity
Information loss
Disclosure risk
Graphs

ABSTRACT

Social networks have become an essential ingredient of interpersonal communication in the modern world. They enable users to express and share common interests, comment upon everyday events with all the people with whom they are connected. Indeed, the growth of social media has been rapid and has resulted in the adoption of social networks to meet specific communities of interest. However, this shared information space can prove to be dangerous in respect of user privacy issues. In addition to explicit "posts" there is much implicit semantic information that is not explicitly given in the posts that the user shares. For these and other reasons, the protection of information pertaining to each user needs to be supported.

In this paper, we present a novel approach wherein the extraction of implicit and explicit information is derived from a small sample of a popular social network (Twitter) that seeks also to preserve user's privacy whilst maintaining information utility.

1. Introduction

Social networks have been adopted massively for the people as a way to communicate between them and, as most sociologists agree, this online interaction will not fade away [1]. People use these networks to share their feelings, emotions, and meet people with the same interests or hobbies. As a result, social networks are plenty of sensible information about each user. Therefore, it can be dangerous to collect this kind of data and publish it without protection. However, this kind of information would be very valuable if published. For example, in [2] we found a case where epidemiology researchers use social networks to study the social structure and epidemic phase in sexually transmitted disease. Usually, in social networks each user has its own public profile where he shows information about himself. However, it should be noticed that there is some information that is not given explicitly but can be inferred such as the user's main topics of interest.

This fact is growing continuously and so, the user's privacy has become a very important issue to deal with. In order to address this issue, data privacy field tries to protect the public data sources (in this case, social networks) allowing the data extraction but taking into account individuals privacy.

There have been several approaches to protect the user anonymity modifying the social graph structure adding and removing edges [3–5]. However, there are less approaches to deal with the privacy in the semantic data included in the graph nodes [6] and approaches to deal with the information that can be extracted from the user's posts [7]. One of the most well known models to protect social graphs is *k*-anonymity which is a very popular model for microdata datasets protection [8], 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 social graph extracted from the real-life Twitter social network [10] using a *k*-anonymity protection method. We wanted to focus on protecting the entire social graph to be able to publish it and, in addition, we wanted to allow the extraction of microdata that is already protected. To do that first we construct a social graph where each node contains a profile describing a single user and is linked to other user profiles. Protection then is done by aggregating the attribute values of user profiles groups following the *k*-anonymity approach. The attributes inside the profiles could be of any type but in this work we focused on categorical attributes because are the most common and the most difficult to protect without loosing a big amount of information.

The main difference between our approach and the others like in [6] is that we do not touch the graph structure and we focus on anonymizing the nodes information as a tool to protect the user's privacy. In this way, our masked graph contains the same amount of nodes and edges than the original graph allowing to perform better studies on it.

Protection methods are typically evaluated using two measures: information loss and disclosure risk (or level of uncertainty)

* Corresponding author. Tel.: +34 93 580 95 70; fax: +34 93 580 96 61.
E-mail addresses: jmares@iiia.csic.es (J. Marés), vtorra@iiia.csic.es (V. Torra).

0950-7051/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2013.05.002

[11]. Information loss checks the quantity of data that has been harmed during the protection process and therefore is no longer useful. Disclosure risk evaluates the information that can be discovered through the protected data.

The remainder of this paper is structured as follows. In Section 2 we explain the methodology we follow to obtain the Twitter social network graph with user profiles containing explicit and implicit information about each user. Section 3 contains the description of the protection method we used to protect the graph's users data. In Section 4 we introduce the measures used to evaluate either the social graph protection and the associated extracted microdata. Section 5 shows some experimental results to check the performance of our protection approach. In Section 6 we make some concluding remarks. Finally, in Section 7 we describe the next steps to do as future work.

2. Social network-extracted graph generation

In this section we present the steps we followed to generate a social graph from real Twitter users connections. These steps are: crawl the Twitter network to extract user's information, generate the users profiles, and construct the graph with these profiles.

2.1. Crawler algorithm

The first step to take is to build a crawler [12] 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 ⇐ uID
actualUser ⇐*getDataFromUser*(id, numTweets)
unvisited ⇐ *getFollowingUsers*(actualUser)
visited ⇐ [id]
Y ⇐ [actualUser]
while (|*unvisited*| > 0) and (|*visited*| < *numUsers*) do
  id ⇐*getRandomId* (unvisited)
  actualUser ⇐*getDataFromUser* (id,numTweets)
  unvisited. *remove* (id)
  newRemaining ⇐*getFollowingUsers* (actualUser)
  unvisited. *add*(newRemaining)
  visited.*add*(id)
  Y.*add*(actualUser)
end while
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 [13] 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 [14], 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 topic 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 [15].

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. This list of three topics is an ordered list according to that frequency.

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 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 construction

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. Fig. 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 are connected 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 represented in the center of the figure, all the users near to him had much more attempts to