Data Retrieval from Online Social Network Profiles for Social Engineering Applications

Sophia Alim, Ruquya Abdul-Rahman, Daniel Neagu and Mick Ridley
Department of Computing, University of Bradford, BD7 1DP
{S.Alim, R.S.H Abdul-Rahman, D.Neagu, M.J.Ridley}@bradford.ac.uk

Abstract

With the increased use of online social networking sites, data retrieval from social networking profiles is becoming a major tool for business. What makes social networking profile data different is its semi-structured format. The structure and the presentation of profile data change all the time. In social networking there is a lack of research into automated data retrieval from semi-structured web pages. Our approach is based on automated retrieval of the profile's attributes and list of top friends from MySpace by examining and extracting the relevant tokens in the parsed HTML code. The tokens were placed into a repository and Breadth First Search algorithm was used. The approach was implemented and tested with a profile which resulted in over 800 top friend profiles and attributes being extracted. This implementation process highlighted that MySpace profile structures vary depending on profile type and the way in which the user has customised the profile.

1. Introduction

The popularity of online social networking sites has increased the amount of personal data which is distributed on the net. This is supported by the fact that social networking sites have overtaken email in terms of usage [2]. Online social networking sites contain user profiles which consist of personal data. Those profiles are semi-structured [18] and the profile data or structure may change in an unpredictable way. This fits in nicely with the way online social networks operate. Social network profiles change all the time not just in structure but content as well. More research needs to be done into the extraction from semi-structured pages in terms of online social networking profiles.

The motivation for this paper is that as far as the paper authors' know there has been little research associated with automated extraction methods from semi-structured web pages from online social networks. Our goal is to extract the relevant profile data so it can be mined in the future to find attributes that can cause the profile owner to be vulnerable to social engineering attacks. In terms of online social networks our research will allow us in the future to investigate the friends of a profile and see if any of the friends have profiles on other online social networks. This links into the transitivity concept where e.g. A and B are friends on one online social network but B and C are also friends on another online social network. The question is: will A and C become friends and if so how great will the strength of their friendship be. The question posed fits with [7] theory about weak ties and how they can provide an alternative information source to the ones associated with the strong ties.

For data to be mined in the first place, it has to be extracted. This paper mainly concentrates on the automatic extraction process of personal details from online social networking sites. Our approach will aim to lower the cost of information retrieval because the attributes from the online social network profiles will be extracted and inserted into a local repository. Data analysis can then take place offline. Due to the fact the online social network profiles in general will change in terms of structure and content on a regular basis, a timestamp can be used to help track changes and these can be followed over time.

The structure of our paper is as follows. Section 1 introduces the problem statement as well as our research contribution. Section 2 explores related work on data extraction from World Wide Web pages, investigating the issues that surround data extraction and analyzing various approaches to combat the data extraction issues. Section 3 presents our methodology in detail and the reasoning behind it. Section 4 discusses the results obtained. Section 5 highlights the conclusions and suggests ideas for future research.

2. Related Work

Data extraction is a field that is concerned with grabbing information from different web resources including websites, online databases and services. It is necessary to find tools for data extraction because of the dynamic nature of the World Wide Web. This

creates some difficulties for end users and application programs when it comes to finding useful data.

There are several issues which prevent finding the pages that the users or applications are seeking for properly. One of those issues is related to information representation. Data on web pages can be found in different formats. HTML is designed for unstructured data which contains information in several formats e.g. text, image, video and audio. It is known that web pages in HTML format are "*dirty*" because their contents are ill-formed and "*broken*" [14]. In contrast, XML and XHTML are designed for structured data. They are stricter in terms of having well-formed documents i.e. the documents' contents should conform to their syntax rules. This feature helps the parsers of search engines to interact with the web pages' contents more efficiently [14]. One of the useful techniques is Wrappers [15] [13] which is responsible for converting HTML documents into semantically meaningful XML files to simplify the operation of extracting data. Wrappers are not efficient though because the programmers have to find the reference point and the absolute tag path of the targeted data content manually. This requires one wrapper for each web site since different sites follow different templates. The effects are increased time consumption and effort from the programmer.

Another issue that prevents efficient data extraction is related to the technique used by search engines to find related web pages. Search engines depend on crawlers to search the World Wide Web for the required keyword(s) that are entered by end users. Crawlers collect information about the visited website then record this information in a process called indexing which is used later on in ranking websites. This problem has arisen because of the limitation of crawlers' capabilities. Crawlers can cover only the Publicly Indexable Web (PIW) while the majority of useful data is in the hidden or deep web, which is not reachable by crawlers [12] [3]. Deep web pages are described by [16] as dynamic pages listing data from databases using a predefined format. Their content is of very high quality since they are managed by organizations interested in maintaining accurate and useful databases.

Extracting data from deep web pages has previously been approached in [16] to deal with drawbacks of some other work tailored to specific web sites. In [19], the approach did not work so well on loosely structured records because they depend on a tree-edit distance metric. The suggested method by [16] avoids those weaknesses by using the Web Data Extractor algorithm which depends on clustering and the weighted tree matching metric to extract data. In [13], Liu and Zhai realized the importance of extracting data records that were retrieved from databases and displayed on Web pages. They analysed the disadvantage of the approaches that were used for extracting data i.e. wrapper induction and automatic extraction then they proposed a method called Nested Data Extraction using tree matching and visual cues (NET) for extracting flat or nested data records automatically.

Since a large amount of information is stored in web databases that are hidden and not indexed by search engines, Hedley et al [9] generated a method that will detect the templates then analyze the textual content and the document's adjacent tag structure to extract query related data. Crescenzi et al [6] demonstrated a system to grab data from the web automatically. The proposed system is also similar to two other authors' approaches: In the first one Kao et al [11] are concerned with analyzing news web sites by identifying pages of news indexes and pages containing news; their work aims to classify pages according to their structure, without any previous assumption. In comparison, Chakrabarti et al's approach [5] is focused on crawling: in contrast to [11], this system is focused on structure recognition rather than searching for pages which are relevant to the topic. This approach does not crawl data behind forms.

Our approach, detailed in section 3, is similar to the work of Park and Barbosa [16], but the difference is that our methodology concentrates more on the structure of the profile and the corresponding tokens. In social networking users can use their imagination when filling in personal details. In [16], using patterns to extract profile data can be strict and therefore it may miss out data that does not fit the pattern. Our methodology (illustrated in Figure 1) proposes extracting attributes and a list of top friends from a MySpace profile. MySpace was chosen because it allows a rich source of data to be derived from profiles without the need to be a member of MySpace. Even if the profile is private, we can still derive some attributes as highlighted below.

3. Methodology

Automated data extraction is just starting to be used in research about online social networking. Table 1 below illustrates some of the data extraction techniques used to extract attributes from online social networking profiles. It shows some data extraction methods ranging from manual through to automated methods. Our methodology (illustrated in Figure 1) outlines our approach to extracting attributes and a list of top friends from a MySpace profile. MySpace was chosen because it allows a rich source of data to be derived from profiles without the need to be a member of MySpace. Even if the profile is private, we can still derive some attributes. Also, when we tried as an external user to extract profiles from different online social network e.g. Facebook and Friendster, either no data or minimal data was made available.

Table 1. Different extraction methods in regards to online social networks.

Method	Research Study	Ref
Developed an automated web crawler using the Ruby programming language. The crawler would visit profile pages based on a randomly generated list of id numbers using the RAND function of Microsoft excel. Regular expressions were used to collect the relevant bits of data.	Age Differences in online social networking	[1]
Wrote two crawlers that were MySpace specific based on "*Perl's LWP User agent and HTML parser modules*". They gathered 2 datasets. One was collected using random sampling and the other one with relationship based sampling.	A large scale study of MySpace Observations and Implications for Online Social Networks	[4]
Downloaded MySpace profiles randomly.	Social Networks, Gender and Friending: An Analysis of MySpace Member Profile	[17]
Used a random number generator to decide which profiles to analyse. Analysis of the profiles took place by manually analysing the profiles and using a data collection form to record their findings	Personal Information of adolescents on the Internet. A quantitative content analysis of MySpace	[10]

3.1. Approach Components

Our process will comprise of the following components:

Stage 1- Data Pre-processing involves analysis of the HTML structure of a given profile. The HTML content is parsed and a vector of tokens is produced. The extracted tokens help in the design of the tables in the repository. The reason of data pre-processing is because MySpace profiles have different structures and therefore different tokens. We created our own MySpace profiles to help investigate different possible structures and attributes.

Stage 2- Specify the URL address of a profile. All social network profiles come with a unique profile URL address. The algorithm for extraction of the personal details involved developing and expanding the library provided in [8]. This code was developed to be applied to online social networking profiles.

We use the URL of the online social network profile as a parameter. Then Java IO methods would be used to extract the HTML of the webpage and store it as a character array. The parsePage method which we defined would remove all the HTML tags from the string, split the remaining text in tokens and place the tokens into a vector. This method proved the most important when extracting the personal details and the list of top friends from the profile.

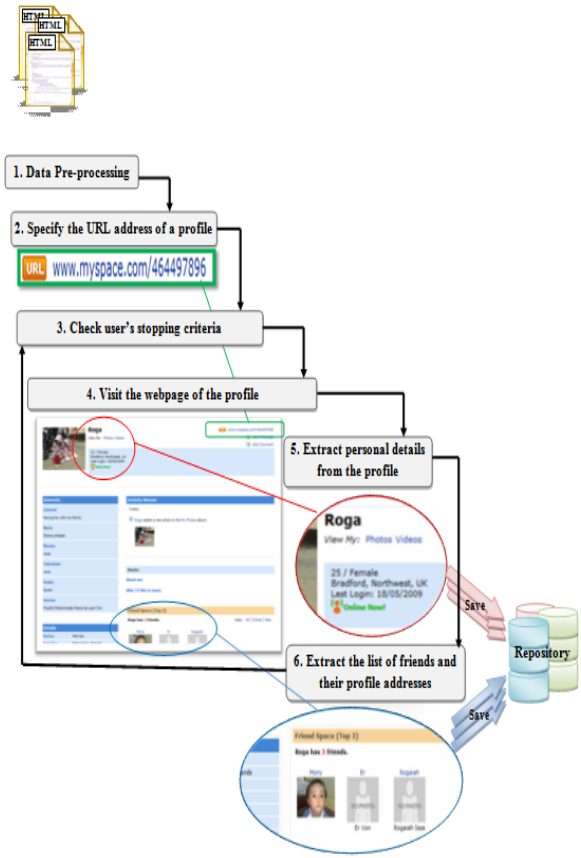

Figure 1. The approach for automated data extraction

Stage 3- Check the user's stopping criteria. The user can specify whether they want to stop the extraction by the number of friends extracted (e.g. the first 100 friends) or by the level (e.g. 1 which is just the top friends of the specified profile) extracted.

Stage 4- Visit the specified profile webpage after checking that it has not been visited before. Breadth First Search has been used for our applications to travel the social network as shown in the algorithm in stage 6.

Stage 5- Extract the relevant personal details from the profile and insert them into the repository. The repository used PostgreSQL 8.1.4. The repository structure has to be designed for the Breadth First Search algorithm.

Stage 6- Extract list of friends and their profile addresses then insert them into the repository if they have not been stored before. In the case of this research paper, the extracted friends' lists just consist of the top friends who we assumed the user may have a strong affiliation with. The data in the repository can be used for data mining purposes in the future to find patterns.

The scenario illustrating Breadth First Search (see Figure 2) will be that profile 1 has 3 friends which are profiles 2, 3 and 4. Profile 2 has three friends: profiles 1, 3 and 5. 5 is a friend of 4. Entrance to the repository is implemented as a queue system:

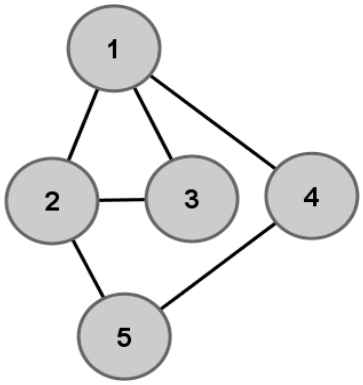

Figure 2. A graph to model Breadth First Search

Using Breadth First Search in this case follows the following steps:

1. Add profile 1's attributes and top friends list into the front of the queue ready to go into the repository.
2. Loop
    a. Look at profile1's friends and check to see if they already exist in the repository. In the first iteration the friends are profiles 2, 3 and 4.
    b. If the friends do not exist in the repository, add their attributes and list of top friends to the rear of the queue.
    c. Look at the next profile at the front of the queue (in this case profile 2) and repeat steps a and b.

3.2. Stopping Criteria

In regards to stopping extraction, we will allow the user to specify the values of two criteria that we set. The criteria are:

- the number of top friends.
- the level of iteration. E.g. a friend of a friend.

In our experiment we stopped the extraction by the level of iteration which in this case was 3.

3.3. Limitations of Current Approach

The limitations of the current approach will provide ideas on what to improve in future research. The limitations include:

1) An incomplete list of friends. The list of friends extracted from the profiles at the moment is not the full friends list. It is just the top friends. A full list of friends will give us a more accurate picture about the environment of the profiles.

2) One mode of travel across the graph. Only Breadth First Search was used to travel across the online social network. This may not be the most efficient algorithm to use so depth first search needs to be implemented as well so we can compare the performance of the two algorithms.

3) Various profile structures. Musicians, magazines and band profiles are not extracted because their profiles were of a different structure. This factor made the profile hard to extract from. Also the friendship between a person and a band differs from that of two individuals who are friends. The friendship between a person and a band is a "*fan based*" relationship compared to a relationship between two individuals which is a "friend of" relationship. A "*fan based*" relationship is less likely to show characteristics of being transitive or reflexive.

4. Experimental Findings

From our experiment we have learnt how to automatically extract data from an online social networking profile using Breadth First Search. The structure of MySpace profiles was found to be different depending on the type of profile and the users' preferences. This proved a challenge when implementing the code. Analysis of HTML structures of various profiles revealed that there was a standard format. Even though some of the profiles were private profiles we could still extract some attributes e.g. nickname, gender, age and location. Data that is placed in the repository can be mined and analysed offline to recognise patterns and trends about the social network in which the profiles are based in.

The profile data can also be used to identify which profile attributes and values make the person vulnerable to social engineering attacks. Vulnerability can be detected by the attributes presented e.g. if the age and horoscope signs are present on a profile then you can have a guess at when the birthday is. If there are comments present on the profile as well you may be able to tell the exact date of the birthday. Other attributes that may contribute to a profile being vulnerable includes whether they are a drinker or a smoker.

5. Conclusions and Future Research

Our research has shown how far social network extraction has come since the days when extracting attributes involved a lot of interaction with the profile owner e.g. questionnaires and interviews. Automatic extraction of attributes is the way forward and it can happen with semi-structured web pages. The main challenge when carrying out the experiment to implement the approach was that social networks like MySpace have more than one profile structure template and the user can customise the template.

This research has provided opportunities for future research to be carried out, as listed below:

1) Extracting the data from the online social networking profiles using a depth first search. The

results from this approach can then be compared to the results from the Breadth First Search. The speed and the amount of memory used will be analyzed to see which searching algorithm would be the most productive in terms of this application.

2) Development of the application to extract all the friends and their attributes from the profile rather than just the top or random friends. This will help to provide a more accurate graph and changes in the online social network can be tracked because the application can be run more than once.

3) Projection of profile connections from the repository into a graph. The graph will map the profiles and their relationships with other profiles. The graph will be a directed weighted multigraph.

4) Development of an agent to automate the process of data retrieval.

5) Run the application over a period of time to track the changes in the online social network and acknowledge the timestamp of current repository content.

6) Extract from other online social networks users with registered profiles.

10. References

[1] R. Arjan, U. Pfeil and P. Zaphiris, "Age difference in online social networking," ACM, Conference on Human Factors in Computing Systems (CHI 08), in *CHI'08 extended abstracts on Human factors in computer systems*, Florence, Italy, 2008, pp. 2739-2744.

[2] BBC, "Social sites eclipse e-mail use," 2009. [Online] available from http://news.bbc.co.uk/1/hi/technology/7932515.stm, last Accessed 4th April 2009

[3] M.K. Bergman, "The deep web surfacing hidden values," 2000. [Online] available from http://grids.ucs.indiana.edu/courses/xinformatics/searchindik/deepwebwhitepaper.pdf last Accessed 23rd April 2009

[4] J. Caverlee and S. Webb, "A Large-Scale Study of MySpace: Observations and Implications for Online Social Networks," AAAI, International Conference on Weblogs and Social Media (ICWSM 2008), in *Proceedings from the 2nd International Conference on Weblogs and Social Media*, Seattle, USA, 2008, pp. 36-44

[5] S. Chakrabarti, M. Van den Berg and B. Dom, "Focused crawling: a new approach to topic specific web resource discovery," *Computer Networks*, vol. 31, no. 11, 1999, pp. 1623-1640.

[6] V. Crescenzi, G. Mecca, P. Merialdo and P. Missier, "An automatic data grabber for large web sites," 30th international conference on very large databases (VLDB 2004), in *Proceedings of the International Conference on Very Large Data bases*, Toronto, Canada, 2004, pp. 1321-1324.

[7] M. S. Granovetter, "The Strength of the Weak Tie: Revisited," Sociological Theory, 1, 1983, pp. 201-233.

[8] Haines, S. *Java 2 from scratch*, QUE, Canada, 1999

[9] Y.L. Hedley, M. Younas, A. James and M. Sanderson, "Query related data extraction of hidden web documents," 2004. [Online] available from http://dis.shef.ac.uk/mark/publications/my_papers/SIGIR2004HedleyYounasJamesSanderson.pdf, Last Accessed 4th April 2009

[10] S. Hinduja and J.W. Patchin, "Personal information of adolescents on the internet: a quantitative content analysis of MySpace," *Journal of Adolescence*, vol. 31, no. 1, 2008, pp. 125-46.

[11] H. Kao, S. Lin and J. Ho and M. Chen, "Mining web informative structures and contents based on entropy analysis," IEEE, *IEEE Transactions on Knowledge and Data Engineering,* vol. 16, no. 1, 2004, pp. 41-55.

[12] Lawrence S and Giles C.L, "Searching the World Wide Web," *Science*, vol. 280, no. 5360, 1998, pp. 98-100.

[13] B. Liu and Y. Zhai, "NET-A system for extracting web data from flat and nested data records," 6th International conference on Web Information Systems Engineering (WISE 05), in *Proceedings of the 6th International Conference on Web Information Systems Engineering*, New York, USA, 2005 pp. 487-495.

[14] L. Ma, G. Goharian and A. Chowdhury, "Automatic data extraction template generated web pages," International conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 03), in *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications*, Las Vegas, Nevada, USA, 2003, pp. 642-648.

[15] J. Palmieri, L. Altigram, S. Da Silva, P.B. Golgher and A.H.F. Laender "Automatic generation of agents for collecting hidden web pages for data extraction," *Data and Knowledge Engineering,* vol. 49, no. 2, 2004, pp. 177-196.

[16] J. Park and D. Barbosa, "Adaptive record extraction from web pages," ACM, 16th International conference of the World Wide Web (WWW2007), in *Proceedings of the 16th International Conference on the World Wide Web,* Banff, Alberta, Canada, 2007, pp. 1335-1336

[17] M. Thelwall , "Social networks, gender and friending: an analysis of MySpace member profiles," *Journal of the American Society for Information Science and Technology*, vol. 59, no. 8, 2008, pp. 1321-1330

[18] J. Widom, "Data Management for XML: Research Directions," IEEE, *IEEE Data Engineering Bulletin, Special Issue on XML*, vol. 22, no. 3, 1999, pp. 44-52

[19] Zhang, K. and D. Shasha, *Tree pattern matching*, ACM, Oxford University Press, Oxford U.K, pp.341-371, 1997