Anonymization Techniques for Data Privacy in Social Networks

G. Prabhakar Raju
Assoc.Prof
Department of Computer Science Engg
Anurag Group of Institutions, Hyd

J Naveen Prasad
Department of Computer Science Engg
Anurag Group of Institution, Hyd

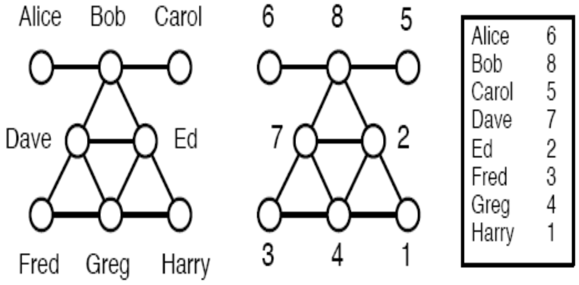
Abstract--The social network data is much more complicated than relational data, privacy preserving in social networks is more challenging and needs many serious efforts in the future. Mainly, modeling adversarial attacks and developing privacy preservation strategies are critical. For proposed work, we believe that the following two types of attacks should be addressed systematically. We only handle 1-neighborhoods in this paper. And another one is k-anonymity. By using sub graphs we can identify the anonymity nodes and for that we are applying anonymity techniques. So we can provide the secure privacy in social network. Finally, privacy research directions on social network sites, privacy-preserving collaborative social network and business model of privacy protection, which Need further research, were presented and discussed.

I. INTRODUCTION

A social network is a social structure made up of a set of actors (such as Individuals or organizations) and the dyadic ties between these actors. The socialnetwork perspective provides a clear way of analyzing the structure of whole social entities. The study of these structures uses social network analysis to identify local and

From at least *k < 1* other records within the same dataset. Global patterns, locate influential entities, and examine network dynamics.

Privacy becomes a more and more serious concern in many applications. The development of techniques that incorporate privacy concerns has become a fruitful direction for database and data mining research. One of the privacy concerned problems is publishing micro data for public use, which has been extensively studied recently. A large category of privacy attacks is to re-identify individuals by joining the published table with some external tables modeling the background knowledge of users. To battle this type of attacks, the mechanism of *k*-anonymity [2] was proposed in. Specifically, a data set is said to be *k*- anonymous (*k* ¸ 1) if, on the quasi-identifier attributes (i.e., the minimal set of attributes in the table that can be joined with external information to re-identify individual records), each record is indistinguishable.



Alice	6
Bob	8
Carol	5
Dave	7
Ed	2
Fred	3
Greg	4
Harry	1

Fig: Edges in social Network.

Neighborhood Attacks

If an adversary has some knowledge about the neighbors of a target victim and the relationship among the neighbors, the victim may be re-identified from a social network even if the victim's identity is preserved using the conventional anonymization techniques. We show that the problem is challenging, and present a practical solution to battle neighborhood attacks. The empirical study indicates that anonymized social networks generated by our method can still be used to answer aggregate network queries with high accuracy.

Privacy [1] becomes a more and more serious concern in many applications. The development of techniques that incorporate privacy concern has become a fruitful direction for database and date mining research.

We are preserving the security attacks on the relational data only. In that type of network we are providing security concerns as authentication.

Proposed Approach

The novel and important problem of preserving [1] privacy in social network data, and took an initiative to combat neighborhood attacks. We modeled the problem systematically and developed a practically feasible approach. Anonymized social networks[4] can still be used to answer aggregate queries accurately. As social network data is much more complicated than relational data, privacy preserving in social networks is much more challenging and needs many serious efforts in the future. Particularly, modeling adversarial attacks and developing privacy preservation strategies are critical. For future work, we believe that the following two types of attacks should be addressed systematically.

II. Social Network Anonymization

One major challenge in anonymizing a social network is that changing labels of vertices and adding edges may affect the neighborhoods of some other vertices as well as the propertiesof the network. It has been well recognized that the following two properties often hold in practical social networks. The properties help us in designing anonymisation methods[1].

NEED FOR ANONYMISING

The problem of protecting against the ability of data recipients to determine sensitive information from other information released to them has been considerably studied in the framework of statistical databases. However, most attention has been devoted to the protection of inference in aggregate statistics and tabular data in contrast to micro data. As a consequence, while a good set of methodologies exist for controlling macro data release “many decisions for the disclosure limitation of micro data are based only on precedents and judgement calls”.

Moreover, most approaches to protect the vulnerability of micro data from linking attacks use technique of data disturbance, which, although safeguarding specific statistical properties compromise the correctness, or truthfulness, of each specific piece of data (tuple in relational terms). Their use therefore limits the usefulness of the released data. The generalization and suppression techniques used in this paper preserve instead information truthfulness, the data released are always correct, although they may be less precise in case of generalization.

Even suppression, although hiding some data, still preserves information truthfulness. In fact, the recipient is informed that some tuples, and how many of them, have been suppressed, and therefore are not released. Such an information can then be taken into consideration by the recipient when processing the released data for producing statistics, in such a way that these statistics will not result distorted. The use of the generalization and suppression as techniques to protect micro data [3] release is not a novelty.

In particular, approaches like recoding (e.g., releasing the month and year of birth instead of the complete birth date), rounding (e.g., rounding incomes to the nearest one thousand dollars), and bottom- and top-coding (e.g., simply releasing the information that a value is smaller or higher than a given amount instead of releasing the specific value) can be represented in terms of generalization. Although generalization and suppression have already been proposed and investigated (often in the combination with other techniques) no formal framework has been provided for their use.

The work closest to ours is represented by two systems, called μ-argus and Data fly, which have also 25 investigated the specific use of generalization and suppression techniques to protect micro data release. The two systems, however, were completely lacking a formalization of the problem and its solutions, which allows us to reason about the correctness and quality of the resulting table. To our knowledge, this paper is the first paper providing that. In some sense therefore, our work and the cited systems represent different levels of looking at the problem and its solutions. We can however attempt some comparison with respect to how generalization and suppression techniques are enforced in these systems and in our model.

In both systems generalization and suppression are used to derive, from a table to be released, a table where combinations of attribute values have, provided some allowed suppression, at least a minimum number (called binsize) of occurrences. The way generalization and suppression techniques are enforced in these systems however presents some shortcomings.

In μ-Argus the user specifies an overall binsize and specifies which attributes are sensitive (i.e., can constitute a quasi-identifier in our terms) by assigning a value between 0 and 3 to each attribute. 2 and 3-combinations across sensitive fields are then evaluated and combinations that have fewer occurrences than the specified binsize are subjected to generalization or suppression. The specified sensitivity values guide the choice of the 2, 3-combinations to be considered. The generalization/suppression process works by considering each sensitive attribute singularly and then 2- and 3- way combinations determined according to the attribute sensitivity. The responsibility of whether to generalize or suppress rests with the user and suppression is enforced at the cell level (for minimizing information loss).

The approach proposed in μ-argus has several drawbacks. The major drawback consists in the fact that despite the binsize requirement, the resulting table may actually allow the recipient to link the information to fewer than binsize respondents. The main reason for this is that μ-argus only checks 2 and 3- way combinations and therefore characterizing combinations composed of more than three fields may not be detected. To illustrate reports an corresponding release from μ-Argus where k=2, the quasi- identifier consists of all the attributes and the weighting of the attributes are as follows. The SSN attribute was tagged “most identifying”; the DOB, Sex, and ZIP attributes were tagged “more identifying”; and the Race attribute was tagged “identifying”.7 Looking at the resulting table: there is only one tuple with values hwhite, 1964, male, 94138i. If this situation is true in the outside world (external table) as well, this individual could be uniquely recognized by the recipient. We can also notice the existence of only one tuple with attribute Sex equal to female and ZIP code equal to 94139. Again, despite the suppression if this is true in the external data as well, also this individual can be re-identified by the recipient.

Data fly also uses the notion of binsize as the minimum number of occurrences of values of an attribute (or combination of them). For each attribute in the table, the data holder specifies an anonymity level, which is a number between 0 and 1 used by Data fly to determine the binsize that the attribute must satisfy. Intuitively, 0 implies no requirement, 1 implies generalization to the maximum level, and the higher the anonymity value the higher the binsize. Each recipient is also assigned a profile composed of a value in the range [0,..,1] for every attribute, describing the probability that the recipient could use the attribute for linking. The binsize to be 6 For the sake of simplicity, ZIP codes have been modified to refer to the values used in this paper. We refer to for details. actually enforced with reference to a specific data release is obtained by weighting the binsize previously specified with the recipient profile. The higher the value in the profile, the higher the binsize requirement. The specification of anonymity and user’s profiles allows flexibility in the anonymity requirement specification, although the actual requirement resulting from a specification may not always be clear. Moreover, control on combination of attributes is executed only for attributes for which a value of 1 is specified for the recipient, so we consider this case for comparison. Given a quasi-identifier and a binsize requirement k computed as discussed above, Data fly cycles on the following basic steps. If the number of outliers is smaller than a specified threshold of suppression, the outliers are removed and the process terminates. Otherwise, one step of generalization is performed on the attribute with the largest number of distinct values. This step is repeat until the tuples achieve the binsize requirement (i.e., k-anonymity)[2] within the suppression threshold. With reference to our framework, Data fly therefore walks through a specific generalization strategy, determined at each step by looking at the occurrences of values in the different attributes, and then stopping at the local minimal solution.

III. PROBLEMS DEFINATION IN SOCIAL NETWORK ANONYMISATION

In this section, we first formulate the problem of privacy preserving in social networks [1] by anonymisation [4]. Then, we review the related work briefly

▪ Identify the Privacy Information to be preserved.

▪ Model the background knowledge that can
be used to attack privacy.

▪Use the anonymisation method to Anonymisation the network ensuring the utilities of the network is not lost.

Identify the Privacy Information to be preserved

In this paper, we are interested in preserving the privacy of individuals which are represented as vertices in a social network. Specifically, how small subsets of vertices are connected in a social network is considered as the privacy of those vertices. Consider a social network *G* = (*V; E; L; L*) and the anonymization *G’* = (*V’;E’;L’;L’*) for publishing. We assume that no fake vertices are added in the anonymization. That is, there is a bijection function *A:V->V’*. This assumption is often desirable in applications since introducing fake vertices may often change the global structure of a social network. Moreover, we assume that for (*u, v*) €*E*, (*A* (*u*);*A* (*v*)) €*E’*. That is, the connections between vertices in *G* are retained in *G’*. For a vertex *u* € *V* , if an adversary can identify a vertex *u’* €*V’*such that how *u* connects to other vertices in *G* is very similar to how *u’*connects to other vertices in *G’*, and is substantially different from how any other vertices connect to others, then the privacy of *u* is leaked. Therefore, privacy preservation in publishing social network data is to prevent any vertex *u 2 V* (*G*) from being reidentified in *G'*with high confidence. Technically, given a positive integer *k*, *G'*preserves the privacy in *G* if every vertex*u* €*V* (*G*) cannot be re-identified in *G'*with a confidence larger than 1/k.

Model the background knowledge that can be used to attack privacy

In order to attack the privacy of a target individual in the original network, i.e., analyze the released anonymisation network and re-identify [3]the vertex, an adversary needs somebackground knowledge. Equipped with different background knowledge, an adversary may conduct different types of attacks against privacy. Therefore,the assumptions of adversaries' background knowledgeplay a critical role in both modeling privacy attacks on socialnetworks and developing anonymization strategies to protect privacy in social network data [3].

In this paper, we assume that an adversary may have thebackground knowledge about the neighborhood of some targetindividuals. This assumption is realistic in many applications. Among many types of information about a target victim thatan adversary may collect to attack the victim's privacy, oneessential piece of information easy to be collected is theneighborhood, i.e., what the neighbors of the victim are andhow the neighbors are connected.

IV. A MODEL FOR ANONYMSING IS'K-ANONYMITY', PROPOSED

In an earlier work, I introduced basic protection models termed null-map, k-map and wrong-map which provide protection by ensuring that released information map to no, k or incorrect entities, respectively. To determine how many individuals each released tuple actually matches requires combining the released data with externally available data and analyzing other possible attacks. Making such a determination directly can be an extremely difficult task for the data holder who releases information. Although I can assume the data holder knows which data in PT also appear externally, and therefore what constitutes a quasi-identifier, the specific values contained in external data cannot be assumed. I therefore seek to protect the information in this work by satisfying a slightly different constraint on released data, termed the k-anonymity[2] requirement. This is a special case of k map protection where k is enforced on the released data.

K-anonymity

Let RT (A1… An) Be a table and QIRT be the quasi-identifier associated with it. RT is said to satisfy k-anonymity if and only if each sequence of values in RT [QIRT] appears with at least k occurrences in RT [QIRT]. L. Sweeney. K-anonymity[2] a model for protecting privacy.

It can be trivially proven that if the released data RT satisfies *k*-anonymity with respect to the quasi-identifier QIPT, then the combination of the released data RT and the external sources on which QIPT was based, cannot link on QIPT or a subset of its attributes to match fewer than *k* individuals. This property holds provided that all attributes in the released table RT which are externally available in combination (i.e., appearing together in an external table or in a possible join of external tables) are defined in the quasi-identifier QIPT associated with the private table PT. This property does not guarantee individuals cannot be identified in RT; there may exist other inference attacks that could reveal the identities of the individuals contained in the data. However, the property does protect RT against inference from linking (by direct matching) to known external sources; and in this context, the solution can provide an effective guard against re-identifying individuals.

The work closest to ours is a forthcoming publication by Backstrom et al. The authors look at social network data that has been naively anonymized, but consider different attacks than those in our work. Their main result regards an active attack, where the adversary does not have knowledge of the graph, but is capable of adding nodes and edges *before* the graph is anonymized. The adversary's strategy is to construct a highly distinguishable subgraph with edges to a set of target nodes, and then to re-identify the subgraph (and consequently the targets) in the published network. The authors provide an algorithm for constructing a subgraph that will be distinguished with high probability.

Much of the work in anonymization has focused on tabular micro data, a database consisting of a single table where each record corresponds to an individual. *K*-anonymity, introduced in, protects tabular micro data against linking attacks in which an adversary uses external information to re-identify individuals. There has been considerable follow-on work on algorithms for constructing *k*-anonymous tables on improving the utility of the anonymized data and on subtler aspects of disclosure, such as inferring properties of the target even without perfect re-identification .

While a graph can be represented in a single table (e.g., an adjacency matrix or edge relation), it does not have the same semantics as tabular micro data (where the records are independent). Applying tabular anonymisation to a tabular representation of a graph will either destroy the graph or fail to provide anonymity.

Recently, there have been extensions of anonymization techniques to more complex datasets. In, the authors present a mechanism, along with strong privacy guarantees, that can be used to share any kind of data, including network data. Rather than publish a perturbed version of the data, they propose an interactive mechanism where the private data is shared with analysts through a privacy-preserving database access protocol. While an interactive approach might be well-suited for some kinds of analysis, it has some disadvantages. The analyst can only ask queries, limiting common social network analysis practices such as visualization and clustering, and he/she can only ask a limited number of queries, limiting data exploration. In, the authors consider anonymizing data in which the individuals are interrelated through employment histories. They use a model-based approach, generating partially synthetic records. The privacy risks are not explicitly considered; the emphasis is on utility, generating replicates that preserve the statistical correlations of the original data.

Achieving privacy through random perturbation has been an area of active work. The author of empirically evaluates the disclosure risk of random perturbation for tabular micro-data, using a framework that is similar to ours. The author observes that while privacy is improved for most records, outliers remain distinguishable. The first perturbation approach that presents a strong guarantee of privacy protection was presented in. The authors consider a scenario where the data collector is untrusted, so the data is perturbed prior to being collected. This technique has been extended to OLAP data. Whether such approaches can be extended to social network data remains an area of future work.

Anonymizing Algorithm

```
Input: a social network $G = (V;E)$, the anonymization requirement
Parameter $k$, the cost function parameters $\alpha,\beta$ and $\gamma$
Output: an anonymised graph $G'$;
Method:
initialize $G' = G$;
mark $vi$ □ $V(G)$ as "unanonymized";
sort $vi$□ $V(G)$ as VertexList in neighborhood size descending order;
WHILE (Vertex List≠Ø) DO
    let SeedVertex = VertexList.head() and remove it from VertexList;
    FOR each vi □ VertexList DO
        calculate Cost (SeedVertex, vi) using the anonymization Method for two vertices;
    END FOR
    IF (VertexList. Size () >=2$k$ -1) DO
        let CandidateSet contain the top $k$ -1 vertices with the smallest Cost;
    ELSE
        let CandidateSet contain the remaining unanonymized vertices;
    suppose Candidate Set= {u1,……..,$u_m$} anonymize Neighbor(SeedVertex) and Neighbor(u1) as discussed in Section III-B.2;
    FOR $j$ = 2 to $m$ DO
        anonymize Neighbor(uj) and fNeighbor(SeedVertex), Neighbor(u1…….,Neighbor(uj-1) g as discussed in Section III-B.2, mark them as "anonymized";
        update VertexList;
    END FOR
END WHILE
```

METHODS FOR ANONYMISATION

Extract the neighborhoods of each vertex

It is much more challenging to model the background knowledge of adversaries and attacks about social network data than that about relational data. On relational data, it is often assumed that a set of attributes serving a quasi-identifier is used to associate data from multiple tables, and attacks mainly come from identifying individuals from the quasi-identifier. However, in a social network many pieces of information
can be used to identify individuals, such as labels of vertices and edges, neighborhood graphs, induced subgraphs, and their combinations. It is much more complicated and much more difficult than the relational case.

Organize the vertices into groups based on similar neighborhoods

It is much more challenging to measure the information loss in anonymizing social network [1] data than that in anonymizing relational data. Typically, the information loss in an anonymized table can be measured using the sum of information loss in individual tuples. Given one tuple in the original table and the corresponding anonymized tuple in the released table, we can calculate the distance between the two
Tuples to measure the information loss at the tuple level. However, a social network consists of a set of vertices and a set of edges. It is hard to compare two social networks by comparing the vertices and edges individually. Two social networks having the same number of vertices and the same number of edges may have very different network-wise properties such as connectivity, betweenness, and diameter. Thus, there can be many different ways to define the measures of information
Loss and anonymization quality.

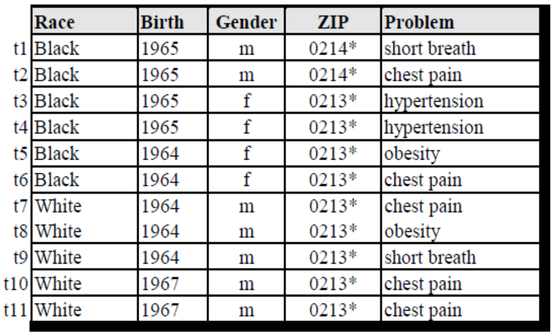
Anonymise the vertices in the same group

It is much more challenging to devise anonymization methods for social network data than for relational data. Divide-and-conquer methods are extensively applied to anonymization of relational data due to the fact that tuples in a relational table are separable in anonymization[1]. In other words, anonymizing a group of tuples does not affect other tuples in the table. However, anonymizing a social network is much more difficult since changing labels of vertices and edges may affect the neighborhoods of other vertices, and removing or adding vertices and edges may affect other vertices and edges as well as the properties of the network.

V. EMPIRICAL EVALUATION

In this section, we report a systematic empirical study to evaluate our anonymization method using both real data sets and synthetic data sets. All the experiments were conducted on a PC computer running the Microsoft Windows XP SP2 Professional Edition operating system, with a 3:0 GHz Pentium 4 CPU, 1:0 GB main memory, and a 160 GB hard disk. The program was implemented in jdk 1.5 and was compiled using Java.

Let RT (*A1*,..., *An*) be a table and *QIRT* be the quasi-identifier associated with it. RT is said to satisfy *k*-anonymity if and only if each sequence of values in RT [*QIRT*] appears with at least *k* occurrences in RT [*QIRT*].



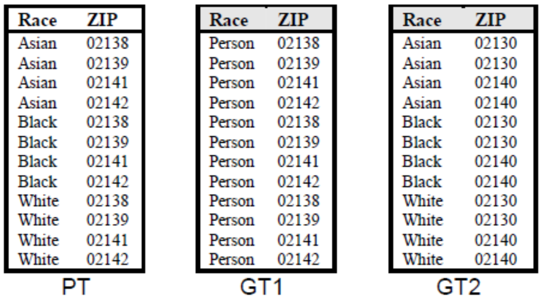
	Race	Birth	Gender	ZIP	Problem
t1	Black	1965	m	0214*	short breath
t2	Black	1965	m	0214*	chest pain
t3	Black	1965	f	0213*	hypertension
t4	Black	1965	f	0213*	hypertension
t5	Black	1964	f	0213*	obesity
t6	Black	1964	f	0213*	chest pain
t7	White	1964	m	0213*	chest pain
t8	White	1964	m	0213*	obesity
t9	White	1964	m	0213*	short breath
t10	White	1967	m	0213*	chest pain
t11	White	1967	m	0213*	chest pain

Fig: Example of *k*-anonymity, where *k*=2 and QI={*Race*, *Birth*, *Gender*, *ZIP*}

Example: Figure provides an example of a table T that adheres to *k*-anonymity [2]. The quasi-identifier for the table is QIT= {*Race*, *Birth*, *Gender*, *ZIP*} and *k*=2. Therefore, for each of the tuples contained in the table T, the values of the tuple that comprise the quasi-identifier appear at least twice in T. That is, for each sequence of values in T[*QIT*] there are at least *2* occurrences of those values in T[*QIT*]. In particular, *t1*[QIT] = *t2*[QIT], *t3*[QIT] = *t4*[QIT], *t5*[QIT] = *t6*[QIT], *t7*[QIT] = *t8*[QIT] = *t9*[QIT], and *t10*[QIT] = *t11*[QIT].

$Cost(u; v) = \alpha.\Sigma_{v'\square H'} NCP(v')+\beta.I\{(v1, v2)I(v1,v2)IE(H),(A(v1),A(v2))\square E(H')\}I+\gamma.(/V(H')I - Iv(H)I)$

It can be trivially proven that if the released data RT satisfies *k*-anonymity with respect to the quasi-identifier QIPT, then the combination of the released data RT and the external sources on which QIPT was based, cannot link on QIPT or a subset of its attributes to match fewer than *k* individuals. This property holds provided that all attributes in the released table RT which are externally available in combination (i.e., appearing together in an external table or in a possible join of external tables) are defined in the quasi-identifier QIPT associated with the private table PT. This property does not guarantee individuals cannot be identified in
RT; there may exist other inference attacks that could reveal the identities of theindividuals contained in the data. However, the property does protect RT againstinference from linking (by direct matching) to known external sources; and in thiscontext, the solution can provide an effective guard against re-identifying[7]individuals.



Race	ZIP
Asian	02138
Asian	02139
Asian	02141
Asian	02142
Black	02138
Black	02139
Black	02141
Black	02142
White	02138
White	02139
White	02141
White	02142

PT

Race	ZIP
Person	02138
Person	02139
Person	02141
Person	02142
Person	02138
Person	02139
Person	02141
Person	02142
Person	02138
Person	02139
Person	02141
Person	02142

GT1

Race	ZIP
Asian	02130
Asian	02130
Asian	02140
Asian	02140
Black	02130
Black	02130
Black	02140
Black	02140
White	02130
White	02130
White	02140
White	02140

GT2

Figure: Examples of k-anonymity tables based on PT

VI. CONCLUSION

In this paper, we tackled the novel and important problem of preserving privacy in social network data, and took an initiative to combat neighborhood attacks. We modeled the problem systematically and developed a practically feasible approach. An extensive empirical study using both a real data set and a series of synthetic data sets strongly indicated that neighborhood attacks are real in practice, and our method is highly feasible. Moreover, anonymized social networks can still be used to answer aggregate queries accurately.

VII. ACKNOWLEDGMENT

The research is supported in part by an NSERC Discovery Grant. All opinions, findings, conclusions and recommendations in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

[1] Bin Zhou, Jian Pei, "Preserving Privacy in Social Networks against Neighborhood Attacks", www.cs.sfu.ca

[2] L. Sweeney, "K-anonymity: a model for protecting privacy," International Journal on uncertainty, Fuzziness and Knowledge-based System, vol. 10, no. 5, pp. 557–570, 2002.

[3] P. Samarati, "Protecting respondents' identities in microdata release," IEEE Transactions on Knowledge and Data Engineering, vol. 13, no. 6, pp. 1010–1027, 2001

[4] M. Hay *et al.*, "Anonymizing social networks," University of Massachusetts Amherst, Tech. Rep. 07-19, 2007

[5] L. Adamic and E. Adar, "How to search a social network," *Social Networks*, vol. 27, no. 3, pp. 187–203, July 2005.

[6] Wikipedia en.wikipedia.com

[7] G. Kossinets and D. J. Watts, Empirical analysis of an evolving social network," *Science*, vol. 311, no. 5757, pp. 88–90, January 2006.