A Synthetic Data Generator for Online Social Network Graphs

David F. Nettleton
Universitat Pompeu Fabra, Barcelona, Spain.

Abstract
Two of the difficulties for data analysts of online social networks are (i) the public availability of data and (ii) respecting the privacy of the users. One possible solution to both of these problems is to use synthetically generated data. However, this presents a series of challenges related to generating a realistic dataset in terms of topologies, attribute values, communities, data distributions, correlations and so on. In the following work we present and validate an approach for populating a graph topology with synthetic data which approximates an online social network. The empirical tests confirm that our approach generates both a diverse and a correlated dataset, with a realistic modeling of noise and interactions between communities. The data generator is also highly configurable, with a sophisticated control parameter set for different 'dispersion' levels.

General Terms
Information and Knowledge, Computational Intelligence.

Keywords
Graphs and networks, online social networks, synthetic data generation, topology, attributes, attribute-values, seeds, communities.

1. Introduction

Online Social Networks have in recent years become of ubiquitous use by people all over the world for social interaction (Facebook) or in business/employment (LinkedIn). In July 2014 Facebook was valued at USD$192bn, its number of users having grown from 1 million in 2004 to 1.32 billion in 2014 (Weil 2015). In 2012, it was estimated that social networks were producing an estimated 2.5 Exabytes of data a day (Mc Afee and Brynjolfsson 2012).

In social networks, users can create sophisticated profiles, defining a rich online set of data about themselves. Also, their activity in the OSNs provides another descriptive dimension of themselves, including friendship links, communication with others, page likes, and so on. However, it is obvious that this data is personal and controls must be applied to protect the privacy of the users. The European Union's recent Data Protection Directive (EU 2015) details legal proposals for the future of how Big Data must be treated. Also, OSNs are susceptible to fraudulent use by the infiltration of fake users, which has been identified as a large scale problem (Kelly 2012). Personalization and user profiling enhances the user experience, but is it well known that user behavior analysis has implications for privacy (Jones et al. 2007; Ramakrishnan 2001).

In the context of research, data analysts who work for the OSN provider companies have a significant advantage with respect to researchers outside of these companies with regards to the access to and analysis of this data; however we assume they must also follow the data privacy legislation in force. Hence, data mining researchers in this area have a serious limitation with respect to data access. One solution would be to conduct specific user studies which would inevitably imply reduced groups of volunteer users who allow a rich set of their OSN data, links and activity to become available for analysis. Another solution is to ask users massively to participate in a study and they volunteer what data they are prepared to make available, in each case. Another solution is to use sampled datasets which guarantee the anonymity of the users and which complies with legal requirements. These are solutions all related to real data. However, in the case of OSNs, simulated data would solve two key problems associated we have mentioned: data availability and data privacy. The option of generating realistic simulated data is the theme for the current work described in this paper.

One issue is how do we know that the simulated data is good or realistic? How can we measure this? Real data also has noise and random aspects, which have to be incorporated. However, we do have tools and definitions within our reach to help us. For example, we can know data distributions for many of the key demographic attribute values in typical OSNs: gender, age, marital status, and so on. Also, we know rules which apply to how people create links with others, based on affinities such as age, gender, residence, education, and so on. This may not give us a perfect match to a real OSN, but it may give us a good approximation which is valid for analysis purposes.

A significant body of research exists in the specialized literature with respect to generator and evolutionary models for topologies (graphs) which represent social networks (Chakrabarti et al. 2004; Lescovec et al. 2005; Robins et al. 2005; Viswanath et al. 2009; Kossinets and Watts 2006; Tang, et al. 2008). However, works on populating these topologies with realistic data are more scarce (Pérez-Rosés and Sebé 2014; Ali et al.2014; Barrett et al. 2009; Boncz et al. 2014) and these are often specific to a given domain or data type.

Hence, in this work our objective is to design and implement a general stochastic modeling system which allows us to populate a graph topology with data, following distribution profiles, attribute value definitions, using a parameterizeable set of data propagation rules and affinities. We benchmark our method with different synthetic and real (ground truth) topologies and the resulting data is evaluated structurally and statistically.

The paper is organized as follows: in Section 2 we describe related work for synthetic topology and data generators; in Section 3 we define some preliminary concepts related to graph topology; in Section 4 we describe our approach for data population of OSN graph topologies; in Section 5 we describe the control parameters for the generator; in Section 6 we present the empirical results and in Section 7 we present the conclusions.

2. Related Work

For convenience, the related work will be divided into two main areas: (i) synthetic topology generation without data and (ii) generating a topology and then generating synthetic data which is then associated with the topology.

2.1 Synthetic Topology Generation

It can be said that the main body of existing work lies in topology generation without data, and a diversity of evolutionary models and generation algorithms exist to produce graph topologies which approximate the characteristics of a real social network (Nettleton 2013). Such characteristics are typically cited as being a small graph diameter, small average path length, skew degree distribution and community structures. (Sala et al. 2010) conveniently divided graph models into three classes based on their approach: feature driven, such as Forest Fire (Lescovec et al. 2005); intent driven, such as random walk and nearest neighbor; and structure driven, such as Kronecker graphs and dK-graphs. A benchmarking was conducted of these different models with respect to their ability to fit to a Facebook graph.

Rmat (Chakrabarti et al. 2004) is a commonly used method which employs a statistical approach and a recursive process to replicate the power law distributions, skew distributions and community structure (which can be hierarchical), while maintaining a small diameter for the graph. The algorithm is optimized in terms of computation cost. Cross-links between communities are also represented. A recursive partitioning is carried out, which can be considered as a binomial cascade in two dimensions. The expected number of nodes c_k with out-degree k is given by:

$$c_k = \binom{E}{k} \sum_{i=0}^{n} \binom{n}{i} \left[p^{n-i}(1-p)^{i}\right]^{k} \left[1-p^{n-i}(1-p)^{i}\right]^{E-k} \tag{1}$$

where ρ is the probability of an edge falling into partition ‘a’ plus the probability of an edge falling into partition ‘b’, and E is the number of edges in the real graph. Also, the number of nodes in the Rmat graph is 2^n, where typically $n = \log_2 N$ and N is the number of nodes in the real graph. Fig. 1 shows a graphical representation of the way Rmat hierarchically processes the dataset. Descriptive parameters are used such as degree distributions, number of reachable pairs, number of hops, effective diameter and stress distribution. One possible deficiency of Rmat is the community structure. (Boncz et al. 2014) and (Pham et al 2013) have reported that the generated topologies have communities with a similar size, instead of the long tail distribution found in real OSNs. However, (Chakrabarti et al. 2004) stated that RMat creates a hierarchical community structure, so a community extraction algorithm would have to take this into account. Also, real OSNs tend not to have neat community boundaries and the real situation is much more fuzzy and overlapping.

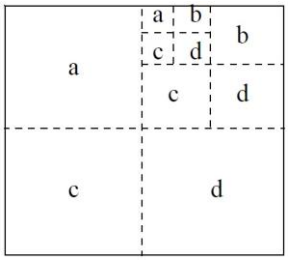

Fig. 1 Graphical representation of RMat hierarchical processing.

(Robins et al. 2005) conducted a simulation for graph sizes ranging from 30 to 500 nodes. For a 100 node graph, up to 500,000 iterations were necessary to reach a stabilization of the statistical values. The model statistics used were: (i) Number of edges; (ii) Number of 2-stars; (iii) Number of 3-stars; (iv) Number of triangles. Aggregate measures (the graph statistics) are then calculated for: (a) Degree distributions; (b) Geodesic distributions; (c) Clustering coefficient. A difficulty was found in the case of the "degree distributions", given that each sample had its own distribution. An "energy" value was defined and calculated for the graph at each iteration, the objective being to find the situation in which the energy reached a minimum. The authors cite four key conditions in order for a small world network to develop: (i) The individuals seek more than one network partner; (ii) The costs of maintaining many partners is high, therefore there is a tendency against a multitude of partners. Dunbar's limit (Dunbar 1993) gives a natural cognitive, sociological and anthropological maximum of 150; (iii) There exists some tendency for network partners to agree about other possible partners, which leads to structural balance and clustering; (iv) If point (iii) is applied in excess this produces cliques with insufficient links between nodes in order to give smaller path lengths. On the other hand, if it is not applied enough there will be insufficient clustering in the network.

A model called "Forest Fire" (with reference to the way link creation propagates), is presented by (Leskovec, et al., 2005). In order to define the model, Leskovec first studied four "social network" datasets over time, in order to see how they change with respect to static models. The datasets studied are 'arXiv citation HEP-TH', 'patents citations', 'autonomous systems (internet routers)' and 'affiliation graph (ArXiv)'. The main conclusions are that the graphs tend to get denser over time, and the diameter tends to shrink, this last conclusion going against 'conventional wisdom'. They define a new graph generator, called the "Forest Fire" model, which is defined by the following: a densification exponent; a difficulty constant; a difficulty function; the number of nodes and edges at time 't'; a community branching factor; the expected average node out-degree; the height of the tree; H(v, w) , which is the least common ancestor height of v, w; the forest fire 'forward burning probability'; the forest fire 'backward burning probability'; the ratio of backward and forward burning probability In terms of structure, the "rich-get-richer" (or preferential attachment) phenomenon is cited as the explanation of the heavy tailed in-degree power-law distribution. Recursive community structures were found for computer networks based on geographical regions. For the patents dataset, the same situation was found in which conceptual groups ("chemistry", "communications", ...) exist. In true OSNs on the other hand, users tend to group together based on "self-similarity". It is noted that in a citation database, a paper only generates outward bound links when it is created. On the other hand, inward bound links will be progressively generated and incremented over time. As a consequence of their observations, the authors require that their model creates a graph with the following characteristics: (i) "rich get richer"; (ii) "copying" which leads to communities; (iii) community guided attachment (densification); (iv) shrinking diameters.

2.2 Synthetic Data and Topology Generation

In contrast to synthetic topology generation, less work exists in building a topology and associated data attributes and values to it. Pérez-Rosés and Sebé (2014) address a specific modeling challenge of simulating skill endorsements between users in LinkedIn and Researchgate. Their approach has two phases: network growth and endorsement modeling via an optimization process.

In (Ali et al.2014; Ali 2014), a synthetic generator is proposed for cloning social network data. They give the example of a dataset extracted from a multiplayer online game, in which nodes represent players, links represent in-game message exchanges, and node features are the player's combat or crafting skills. They state that social media datasets can exhibit profile homophily, that is, an increased likelihood of connection between users with similar profiles. Their generator includes dynamic label homophily fitting, attribute assignment and optimization and link formation based on feature similarity. For the attribute assignment optimization they tried particle swarm optimization and genetic algorithms. They evaluated

their methods on three datasets: DBLP-A, Travian and GameX. They were able to obtain a very close fitting for these datasets, at the cost of computational time and resources.

(Barrett et al. 2009) consider the generation and analysis of large synthetic social contact network. The work of (Barrett et al. 2009) is somewhat different from the previous ones we have cited so far, because their objective is to model a countries population based on census type demographic and household information. This model is then used to predict spread of epidemics based on physical co-location (geographic proximity) and several large cities are benchmarked such as New York, Los Angeles and Seattle. A key aspect of the model includes trip/journey behavior of individuals based on their employment, population and household densities. They use labeled bipartite interaction graph to captures visits by people to different locations. Age group is also a key factor. They use: pre (school (<5), school-age (5-18), adults (19-64) and seniors (>65).

An approach whose origins lie in the graph database field is that of (Boncz et al. 2014) and (Pham et al. 2013). It is oriented to the performance evaluation of "choke points" queries with a high computational cost, and of returning realistic results from SQL type queries by creating local neighborhood based primarily on demographic data affinities.

In previous work (Nettleton 2014; Nettleton 2015) an initial version of a synthetic OSN data generator was described for non overlapping communities using an RMat (Chakrabarti et al. 2004) generated topology and simple control parameters. In (Nettleton 2014) the synthetic data was used in a data privacy application.

3. Preliminaries

A graph G is defined as a set of vertices V interconnected by a set of edges E, denoted by G = (V, E).

In this work, for modeling social interactions we assume that the graph is a weighted graph, that is for each edge e it has associated a numerical weight value $w(e) \in [0,1]$ which is an indicator of the strength of relation (e.g. interaction intensity).

We consider the weighted graph G together with a table T, in which each tuple corresponds to a vertex v and has $\{a_1, a_2, a_{3, \dots}, a_n\}$ as attributes and $\{va_1, va_2, va_{3, \dots}, va_n\}$ as corresponding values.

The complete graph G is subdivided into communities $c \in C$, labeled by the Louvain method or by the ground truth communities.

In order to avoid overlap/overwriting in the assignment of the data we use a set of seed vertices that are going to be chosen with the following properties: each seed has to have distance at least 2 to all the other seeds; each vertex of the original graph G is at distance at most 2 to some seed vertex; the seeds are chosen from the list of nodes in a community c, ordered by their distance to the medoid node of c M_c as calculated by the centrality metric. The medoid M_c and centrality metric facilitate a homogeneous and optimum distribution of seed throughout the community topology. It is a natural assumption that the OSN graphs have to be similar between close acquaintances, hence, the condition of having a seed vertex at distance at most 2 guarantees that the vertices that are out from the set of seeds are at distance at most one from some seed's neighbor and therefore will intuitively be well represented.

Denote the set of seed vertices for a given community as as $S_c=\{sc_1, sc_2, \dots, sc_n\}$.

We denote the closed neighborhood of a seed vertex $s \in V(G)$ by $N(s)$ and it consists of all the neighbors of s in G together with s and all the edges of G that connect them. The neighborhood is a key aspect of the data propagation, given that a seed is assigned a profile directly, whereas its neighbors which are in the same community as the seed will be assigned a profile with a 'similarity' to that of the seed, as determined by the control parameters described later in Section 5.

Also, a set of profiles P is defined. Each profile P has attributes $a \in A$ and each attribute has a value $a_v \in A_V$ assigned from those defined for the given attribute.

For a seed vertex sc_i, our data assignment method chooses the seed vertices $sc_2, \dots, sc_n$ such that $M_c - sc_i$ is a minimum. These seeds are the ones to be assigned the predefined data profiles $P_1, \dots P_c$.

4. Description of the Method

The method has three overall steps which will be described in the following: topology generation/definition, data definition and data population. It could be debated that the data definition step should come first, followed by the definition of the topology, or that the topology should be evolved together with the data generation, such as in (Boncz et al. 2014). However, in the present work, our focus

and contribution is the population of an already existing topology, such as the ground truth graphs we benchmark in Section 6. The topology generation/definition has two options: (i) synthetic topologies generated by RMat and then community identification using the Louvain method; (ii) use of real topologies in which the "ground truth" communities are already identified.

Once we have the topology and the communities assigned we define the data we wish to use. We define the attributes and their values, together with the general percentage frequency in the complete population for each attribute-value. Next we define a set of distinctive profiles in terms of the attribute-values described previously. For each profile we assign a target frequency which indicates the desired percentage of the records which will have this profile. The last step is to populate the empty topology with data, using the attribute values and the profiles defined previously. This is done by assigning "seed" nodes in each community and propagating data to their immediate neighbors and beyond until all the nodes in the graph have data assigned.

4.1 Step 1: Topology Preprocessing

Firstly, we have to obtain a topological structure. In the current work we have applied two contrasting approaches. On the one hand we have used RMat to generate synthetic topologies, and on the other hand we have obtained topologies of real OSN community ground truths from the SNAP online repository (Amazon, Youtube and LiveJournal).

For the RMat generated graph, we identify the communities in the graph structure by processing with the Louvain method (Blondel et al. 2008), which assigns a community label to each vertex in the graph. We note that we consider that the communities of the RMat model are non-overlapping.

For the ground truth graph datasets the (real) community is already identified, although we need some reformatting in order to obtain the required input files. In the case of the real graphs, the communities were overlapping, with a node potentially being a member of many communities. For each dataset we chose the reduced (5000 top communities) option.

RMat generated graph: in Table 1 we see the size of each community as a percentage of the total nodes in the whole graph. We note that when we applied the Louvain method to the RMat generated topology, it tended to obtain communities of a similar size. For the 1K nodes graph, six communities were extracted by Louvain with the optimum modularity, corresponding to Ids 0 to 5 in Table 1, and with a size of between 17% and 22% of the complete graph. Thus, in order to obtain some resemblance to a "long-tail" distribution of the community sizes, which is more typical of a real online social network, we applied the Louvain method recursively to communities 3 and 4 (which had the highest modularity values). From the resulting sub-communities we chose the biggest and the smallest. For community 3 this gave us communities 6 and 7, and for community 4 this gave us communities 8 and 9. We note that the resulting communities, 3, 6 and 7 are non-overlapping. The same applies to communities 4, 8 and 9.

Table 1. RMat graph: communities and their % size with respect to the whole graph.

Community Id	0	1	2	3	4	5	6	7	8	9
% of whole graph	0.216	0.172	0.211	0.097	0.081	0.157	0.024	0.005	0.028	0.009

Finally, we identify a set of seed vertices which will be used as the starting points for propagating the data. For each community, we find the medoid node in terms of the statistical and topological characteristics (especially centrality and degree). Then we progressively assign seed vertices whose characteristics are closest to the medoid in each community, which gives a close to optimal coverage of the complete graph. We note that a rule was applied in which the neighbors of a seed node must be disjunct from the neighbors of any other seed node in the same community. This prevents overlapping of their immediate neighborhoods which avoids overwriting data propagated from different seeds. We were able to assign 110 seed nodes in this manner for the 1K RMat generated graph.

In the case of the ground truth datasets, a similar procedure was followed, except that multiple community membership was taken into account. That is, if a seed node is a member of twenty communities, it is assigned a data profile just once and the assignment is registered for all the communities in which it is a member.

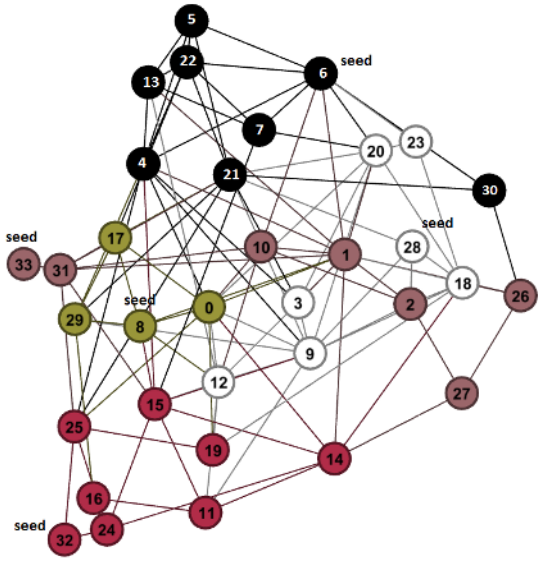

Fig. 2: Example of Rmat generated topology. Different colors indicate communities. Seed nodes are indicated by 'seed'.

4.2 Step 2: Data Definition

The choice of data will be application specific. However, the distributions of the values of the different attributes should be similar to that of a real social network (ground truth). In order to achieve this, we can use sources of official statistics, such as government census data (*www.indexmundi.com*, *www.census.gov*, *www.bls.gov*), and statistical summaries made public by the social network providers, such as Facebook (*www.adweek.com*, *fanpagelist.com*, *http://royal.pingdom.com/2009/11/27/study-males-vs-females-in-social-networks/*). Example attributes and a-priori proportions are shown in Table 2. We may also need some lookup tables for highly inter-related attributes values. For example, for the age group "18-25", there will be a much higher proportion of "profession=student" and "marital status=single", and for "gender=male" there will be a higher proportion of "like3=soccer club". Another option for 'ground truth' are publicly available OSN datasets, such as those found at the SNAP website: *http://snap.stanford.edu/data/#communities*.

Table 2. Example attributes, attribute-values and their demographic (US Census, 2010) proportions.

Attribute	Values
Age	"18-25" (25%), "26-35" (25%) , "36-45" (16.67%), "46-55" (8.33%), "56-65" (8.33%), "66-75" (8.33%), "76-85" (8.33%)
Gender	male (47%), female (53%)
Residence	"Palo Alto" (17%), "Santa Barbara" (16%), "Boca Raton" (16%), "Boston" (17%), "Norfolk" (17%), "San Jose" (17%)
Religion	"Christian" (31.9%), "Hindu" (14.8%), "Jewish" (0.2%), "Muslim" (27.1%), "Sikh" (0.3%), "Traditional Spirituality" (0.1%), "Other Religions" (12.9%), "No religious affiliation" (12.7%)
Marital status	"Single" (31.5%), "Married" (51.4%), "Divorced" (10.5%), "Widowed" (6.6%)
Profession (ISCO-08 structure)	"Manager" (12.2%), "Professional" (17.1%), "Service" (13.9%), "Sales and office" (17.8%), "Student" (23%), "Natural resources construction and maintenance" (7.0%), "Production transportation and material moving" (9.0%)
Political orientation	"Far Left" (9.4%), "Left" (34.7%),"Center Left", (18.1%), "Center" (18.0%), "Center Right" (10.5%), "Right" (8.0%), "Far Right" (1.3%)
{like1, like2, like3}	Patterns: {"entertainment", "entertainment", "music artist"} (25%), {"music artist", "music artist", "entertainment"} (25%), {"drink brand", "drink brand", "entertainment"} (25%), {"tv show", "drink brand", "soccer club"} (25%).

Table 3. Example profiles and their overall target proportions for the complete dataset.

Profile Id	Profile	Target %
0	36-45, Male, Boston, No religious affiliation, Married, Sale and office, Center, Heterosexual, TV show, Drink brand, Soccer club.	0.216
1	26-35, Female, Cambridge, Buddhist, Divorced, Manager, Left, Bisexual, Music artist, Music artist, Entertainment.	0.211
2	18-25, Male, Palo Alto, Christian, Single, Student, Center left, heterosexual, drink brand, drink brand, entertainment.	0.172
3	18-25, Female, Winthrop, Muslim, single, professional, center right, heterosexual, entertainment, entertainment, music artist.	0.157
4	56-65, Male, Santa Barbara, Hindu, widowed, Natural resources construction and maintenance, Right, Heterosexual, TV show, drink brand, soccer club.	0.097
5	66-75, Female, San Jose, Jewish, married, Production transportation and material moving, far left, heterosexual, music artist, music artist, entertainment.	0.081
6	18-25, Female, Winthrop, Christian, Single, Professional, center right, heterosexual, entertainment, entertainment, music artist.	0.028
7	18-25, Female, Winthrop, Jewish, Single, professional, center right, heterosexual, entertainment, entertainment, music artist.	0.024
8	56-65, Male, Santa Barbara, Hindu, widowed, Natural resources construction and maintenance, left, heterosexual, TV show, drink brand, soccer club.	0.009
9	56-65, Male, Santa Barbara, Hindu, Widowed, Natural resources construction and maintenance, Far Left, Heterosexual, TV show, drink brand, soccer club.	0.005

Two specific sub-steps for data definition are as follows:

1. Define each attribute; define possible values for each attribute; define percentage of total population which have each attribute-value. (Table 2).

2. Define data profiles (see Table 3).

RMat: In the case of the RMat dataset, we define one profile for each community extracted by Louvain (we note that the number of communities can be controlled/limited by an input parameter). We also define what percentage of total dataset is desired (Target %) for each profile. The profiles are detailed in Table 3. Each profile will then be matched with the community whose percentage of the complete graph (in terms of number of nodes) is closest to the desired percentage for the profile, and assigned to its seeds.

Ground Truth datasets: these datasets have a much larger number of communities (5000) which display a long tail size distribution. Hence, we have a fixed number of profiles (for example, 10) and we define an assignment probability to each (equivalent to the Target % used for the RMat dataset). Then, the seeds in the communities will be pseudo-randomly assigned the profiles depending on the assignment probability. For example, Profile 2 has a Target % of 21.1, thus Profile will be chosen, on average, 21.2% of the time to be assigned to the seeds of a given community.

4.3 Step 3: Data Population

The four specific sub-steps for data population are as follows:

1. Assign each profile prototype to seeds of corresponding community. For Rmat generated topology, match profile percentages (Table 3) defined by user to community percentages present in topology. For ground truth topologies, assign profiles with a probability proportional to the target percentage (see Table 3)

Table 4. Rmat Topology. Assignment of profiles to communities based on their desired (profiles) and calculated (communities) % size with respect to the whole graph.

Community Id	0	1	2	3	4	5	6	7	8	9
Assigned Profile	0	2	1	4	5	3	7	9	6	8

2. Assign neighbors of seeds based on profiles. Each neighbor attribute has a maximum allowed distance from corresponding seed attribute of z%. Neighbor attributes are assigned randomly k% of the time.

3. Assign attributes of nodes still unassigned (which are neither seeds nor neighbors of seeds). For each node, p% of the time a random assignment (by default p=10) and q% of the time (90% by default) each attribute is assigned the modal value of the neighbors of the node.

4. Check statistics of communities and whole graph. If not within desired limits, return to previous steps and modify control parameters (see Section 5).

To initiate the population of the network with data, we use the set of seed nodes mentioned previously. The rest of the nodes will be assigned data by propagating from the seed nodes. The immediate neighbors of a seed will have a higher probability of being assigned similar attribute values. We also use ontologies /taxonomies and a distance measure to assign similar, rather than identical values (with an appropriate threshold) when propagating attribute-values.

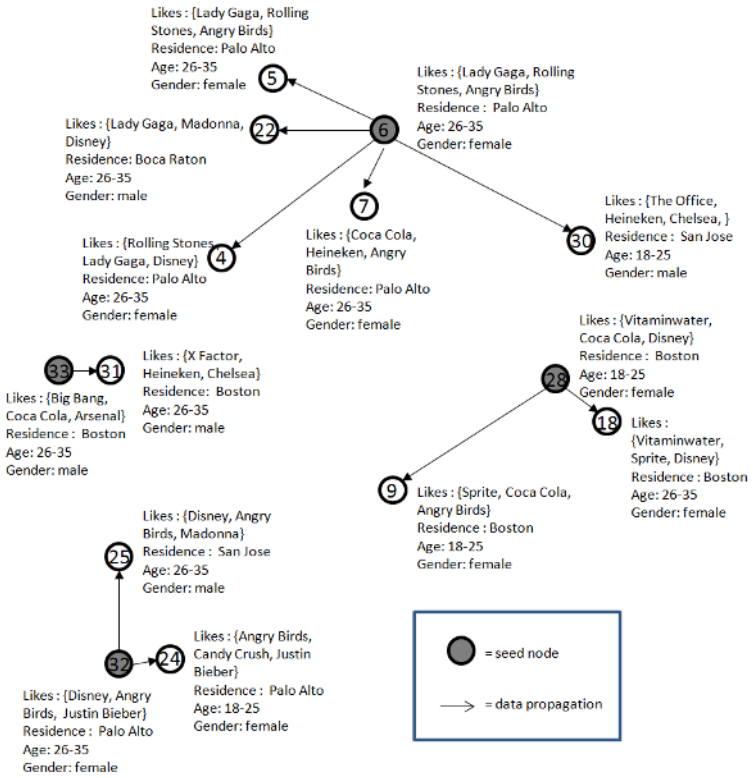

Fig. 3: Data propagation from seed nodes to immediate neighbors, in the topology of Fig. 2.

The influence on assignment by the seed node has to be traded off by the desired overall proportions of the attribute values (diversity). In order to optimize the assignment, we can use a fitness function and find the optimum configuration for the control parameters using a stochastic process.

For example, Fitness = $f(\Omega, \Phi, \Omega)$, where Ω = set of seed assignments, Φ = set of data propagation probabilities and distance thresholds, Ω = set of required data distributions (profiles).

5. Control Parameters

In this section we will describe how we can control the generator behavior by define a set of control parameters which can act generally and also on specific attributes and their possible values. In order to change the resulting distributions and assignments, we can vary these parameters. There are five major control parameters, which are: NS, number of seeds; {SP}, set of seed profiles; PR=probability ranges; {DT} set of distance thresholds for each attribute; RU, random assignment threshold. Each of these parameters will now be described in detail.

(i) NS, Number of seeds (110 for Rmat 1000 node graph, 5000 for Amazon, 12000 for Youtube and LiveJournal). This value is dataset dependent and by several trials on each graph dataset we found an optimum value in terms of processing time and coverage (evaluated as the number of nodes during processing which were neither seeds nor neighbors of seeds).

(ii) SP, Seed profiles. RMat: same number as communities, % desired for each. Ground truth datasets: probabilistic assignment of profiles to communities. Examples are shown in Table 3.

(iii) PR, Probability Range which controls the assignment of different distance thresholds (DTs) for the attribute-values of seed neighbors. In general, three ranges are used: from zero to *x*, *x* to *y* and *y* to 1. '*x*' is typically assigned a value of 60 and *y* is typically assign a value of 90. This gives the result that 60% of the time the neighbors are assign identical attribute-values as the seed; 30% of the time they are assigned attribute-values different but within the closest distance threshold; and 10% of the time they are assigned attribute-values whose distance is greater than the closest distance threshold. Reducing the first range and proportionately increasing the other two will make the community more varied, with a greater dispersion of attribute-value distribution. Enlarging the first range will make a higher proportion of the neighbors have the same attribute-value to their corresponding seed.

So, we can define three default assignments for the probability ranges: "low" = {60, 30, 10} will give a low dispersion; "medium" = {50, 25, 25} will give a medium dispersion; "high" = {40, 30, 30} and will give a high dispersion.

(iv) DT, Distance Threshold. This defines the similarity between a seed attribute-value and a neighbor attribute value. Set for each attribute. The minimum distance between seed and neighbor which has to be achieved for each attribute-value. The distance range is attribute dependent, because the nature of the values affects how we calculate the distance. See Table 2 for a complete list of attributes and their possible values. In general the first distance threshold is zero, which means the neighbor will be assigned the same attribute-value as its seed; the second distance threshold is equal to the distance to the most similar distinct attribute-value. For politics and age, its 1/6, for gender its 1, for religion, sexual orientation and marital status its ½. For likes, two thresholds are used, 0.15 and 0.24. Residence has two thresholds, ¼ and ½.

Let's take the politics attribute and its values as an example. Consider the attribute-value politics=center. The two closest values to "center" for attribute politics are "center left" and "center right", which are at distance 1/6 from "center". Next we have "left" and "right" which are at distance 2/6 from "center". Finally we have "far left" and "far right" which are at distance 3/6. So, we can define threshold 1 to be 0 and threshold 2 to be 1/6. Next we define for each threshold, the probability range. As we described in the previous step (PR), we set PR so that 60% of the time we assign attribute-values with distance zero (neighbor attribute-value same as seed attribute-value), which will be "center". We also set PR so that 30% of the time, we assign attribute-values with distance 1/6 (neighbor attribute-value is at distance 1/6 from seed attribute-value), which will be "center-left" or "center-right". Also we set PR so that 10% of the time, we assign attribute-values with distance > 1/6 (neighbor attribute-value is at a distance greater than 1/6 from seed attribute-value), which will be one of "left", "right", "far left" or "far right".

Table 5. Allowed distance ranges for attribute-value assignment (seed to neighbors) in a community

Closest Distance thresholds								
age	**gender**	**residence**	**politics**	**sexuality**	**religion**	**marital**	**profession**	**likes**
1/6	1	1/4	1/6	1/2	1/2	1/2	1/2	0.15

In practice, we keep the distance thresholds constant and use the PR's to vary the proportions of attribute-values whose distances are closer to or further from the seed attribute values.

Attribute "gender" is a nominal and has two possible values and the distance between two instance will be 0 (the same) or 1 (different).

Attribute "age" is an ordinal and has 7 possible values (categories). The distance goes from 0, in steps of 1/6 to 1. Attribute "political orientation" is also considered as ordinal and has 6 possible values (categories, see Table 2). The distance goes from 0, in steps of 1/6 to 1.

Residence is represented as a hierarchical category with four geographic levels (United States): county, state, division and region. If the residence of two instances is equal the distance is zero; if the residence is not equal but it is in the same county, the distance is 0.25; if the residence is not equal but it is in the same state, the distance is 0.50; if the residence is not equal but it is in the same division, the distance is 0.75; if the residence is not equal but it is in the same region, the distance is 0.90; otherwise the distance is 1.0.

The distance between likes is calculated using an "affinity" table. When two likes are the same, the distance is zero, otherwise the pair is looked up in the table to find the corresponding affinity. The distance between "entertainment" and "music artist" is 0.25; between "music artist" and "entertainment" is 0.25; between "tv show" and "drink brand" is 0.50; between "soccer club" and "drink brand" is 0.50; and so on.

When we calculate the distance we sum the distance of each seed value to each neighbor value. For example, consider the case when the seed 'like' values are: sv_1='entertainment', sv_2='entertainment' and sv_3='music artist' and the potential neighbor 'like' values are: nv_1='music artist', nv_2='music artist' and nv_3='entertainment'. Firstly we compare and calculate the distance of sv_1 to nv_1, nv_2 and nv_3 which gives corresponding distances of 0.25, 0.25 and 0.00 and a subtotal of 0.50. Now we do the same for sv_2 calculating its distance to nv_1, nv_2 and nv_3, which again gives 0.25, 0.25 and 0.00 and a subtotal of 0.50. Next we do the same for sv_3 calculating its distance to nv_1, nv_2 and nv_3, which gives 0.00, 0.00 and 0.25 and a subtotal of 0.25. Lastly we sum the three subtotal to give the distance between the tuples $\{sv_1, sv_2, sv_3\}$ and $\{nv_1, nv_2, nv_3\}$ as 0.50 + 0.50 + 0.25 = 1.25. Finally, we divide by 9 to give a normalized value (between 0 and 1) of 0.139.

For the attribute religion we have considered that "Buddhist", "Hindu" and "Sikh" have a relative affinity so if the religion for each of two instances is not equal but is one of these three, then their mutual distance will be 0.5. In a similar manner, if the religion for two instances is not equal but is one of "Christian" or "Jewish", then their mutual distance will be 0.5. Otherwise, the distance will be zero (different) or 1 (equal).

For the attribute marital status we have considered that "Married", "Divorced" and "Widow" have a relative affinity so if the marital status for each of two instances is not equal but is one of these three, then their mutual distance will be 0.5. Otherwise, the distance will be zero (different) or 1 (equal).

For the attribute profession we have considered that "Manager" and "Professional" have a relative affinity so if the profession for each of two instances is not equal but is one of these two, then their mutual distance will be 0.5. In a similar manner, if the profession for two instances is not equal but is one of "Service" or "Sales and office", then their mutual distance will be 0.5. The same applies for the professions "Natural resources construction and maintenance" and "Production transportation and material moving". Otherwise, the distance will be zero (different) or 1 (equal).

Finally, we define a weight w which is assigned to each edge e (link between two user nodes in the graph) where $w(e) \in [0,1]$ which is an indicator of the strength of relation (e.g. interaction intensity). This is calculated as the last step of the processing, when all the data is assigned. The value of the weight is calculated as the 'grade of similarity' or 'distance' between the respective attribute-value sets of two user nodes. The distance between each attribute-value is calculated in the same manner as we have described in this Section for the distance thresholds (DT). The overall distance is given by the weighted sum of the attribute values, where an equal attribute weighting is used by default.

We note that it is clear that the rules we have defined are modifiable depending on the data, context and application.

(v) RU, Random assignment threshold for unassigned nodes. Nodes which are neither seeds nor neighbors of seeds can have their attribute-values assigned randomly or they can be assigned as the mean/modal values of their neighbors which have already been assigned attribute-values. An example threshold would be 10%, that is, 10% of the time the assignment is random and 90% of the time the assignment is based on the modal values. Making the threshold bigger will make the community less homogeneous and less similar to the seed profiles. This may be useful in the case that we wish to control the overall distributions of minority attribute values. For example, in the current overall distribution we have a relatively high proportion (approx 17% of "religion=Buddhist". If we wish to make the overall distributions representative of, for example, the United States, we would have to reduce this overall proportion. This could be done by increasing the random assignment for the corresponding profile/community.

(vi) Control Parameter Set $\mathbb{C}P$. A first control parameter set is defined as [NS, {SP}, PR = {60, 30, 10}, {DT}, RU=10%]. This corresponds to a lower intra-community diversity and a high correlation between nodes and their neighbors. We designate this as "Level1".

A configuration for a somewhat higher dispersion than Level 1 would be [NS, {SP}, PR = {50, 25, 25}, {DT}, RU=30%]. We designate this as "Level2".

A configuration for a somewhat higher dispersion than Level 2 would be [NS, {SP}, PR = {40, 30, 30}, {DT}, RU=50%]. We designate this as "Level3".

In Table 6 we see a summary of the assigned ranges for the three levels of dispersion we have defined and tested.

Table 6. Control Parameters $\mathbb{C}P$ for seed to neighbor and non-neighbor assignment.

	R† %			RU‡ %	
Dispersion Level	**Same**	**Close**	**Other**	**Medoid Neighbors**	**Random**
Level 1 - Low	60	30	10	90	10
Level 2 - Medium	50	25	25	70	30
Level 3 -High	40	30	30	50	50

†Probability ranges for distance assignment to neighbors; ‡Random assignment percentage for unassigned nodes

After a number of trials with different graphs datasets, number of nodes and communities, we found that the Level2 dispersion control parameter set gave the best results. A good result was evaluated as being that the communities had a clearly identifiable profile but a realistic diversity and noise was also present in the attribute values of each record. It was found that obtaining a good dispersion was also dependent on the community size, where smaller communities tended to have homogenous assignments (because a seed would be connected to all members of the community).

6. Empirical Analysis

In this section we present the statistical evaluation of the generated OSN graphs and associated data. First, in Section 6.1 we analyze the data for which RMat has been used to generate the topology. This topology is designed to have the same number of communities as profiles. Each user is a member of only one community. Then in Section 6.2 we analyze the data for which three real ground truth topologies are used. These topologies have a large number of communities (5000) and users can be members of many communities. We note that the 'likes' and the 'edge weight' attributes have not been included in the analysis of the results due to space restrictions and to maintain clarity. We note that the proportional distributions of the 'like' attribute-values followed a similar tendency to the other categorical values, as expected. We also confirmed that the 'edge weight' values were correlated to the similarity between attribute-value sets of corresponding connected node (user) pairs, as defined in Section 5 (iv).

6.1 RMat generated topology

In the following we present the results for the RMat generated 1K node graph in which the communities were identified/labeled by the Louvain method.

6.1.1 Data Distributions: matching of profiles to communities

Fig. 4 shows the attribute-value distributions for the whole graph. We see that age is predominantly categories 18-25 and 26-35, gender is equitably distributed and sexual orientation has a high imbalance. The last attribute (community id) has been defined as the "class value" and thus all other attribute-values show the composition with respect to this attribute.

For example, community 2 (light blue) has a high relative proportion of attribute values "gender=female", "religion=Buddhist", "residence=Cambridge", "profession=Manager", "sexual orientation=bisexual" and "marital status=divorced". If we check the community→profile correspondence, we find that community 2 has been assigned profile 1, and profile 1 includes these attribute value assignments. Likewise, community 3 (grey/blue) has a high proportion of attribute-values "gender=male", "religion=Hindu", "residence=Santa Barbara", "profession=Natural…" and "marital status=widow". If we check the community→profile correspondence, we find that community 3 has been assigned profile 4, and profile 4 includes these attribute value assignments. Fig. 5 shows the attribute-value distributions only for community 3. The bias of the attribute-value distributions to profile 4 is clearly evident. Hence, we have successfully obtained the desired characteristics for this community. Table 7 shows the distributions for the values of attributes 'age', 'religion' and 'profession' for each community. The bias of the attribute-

value distributions to the corresponding profiles (see Tables 2 and 3) is clearly evident. Hence, we have successfully obtained the desired characteristics for this community.

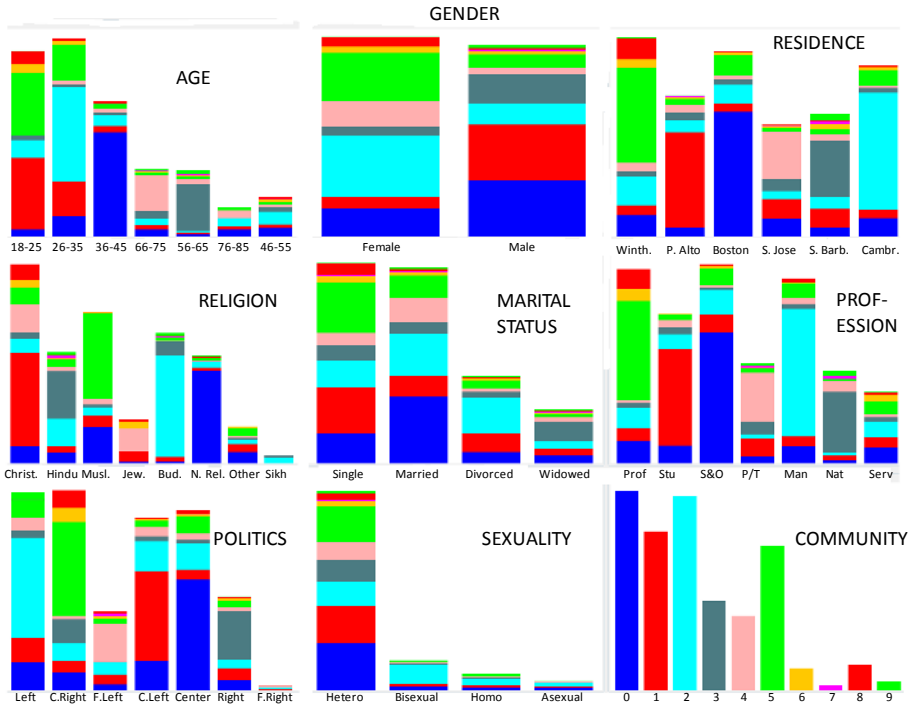

Fig. 4: Distributions of attribute-values for the complete graph.

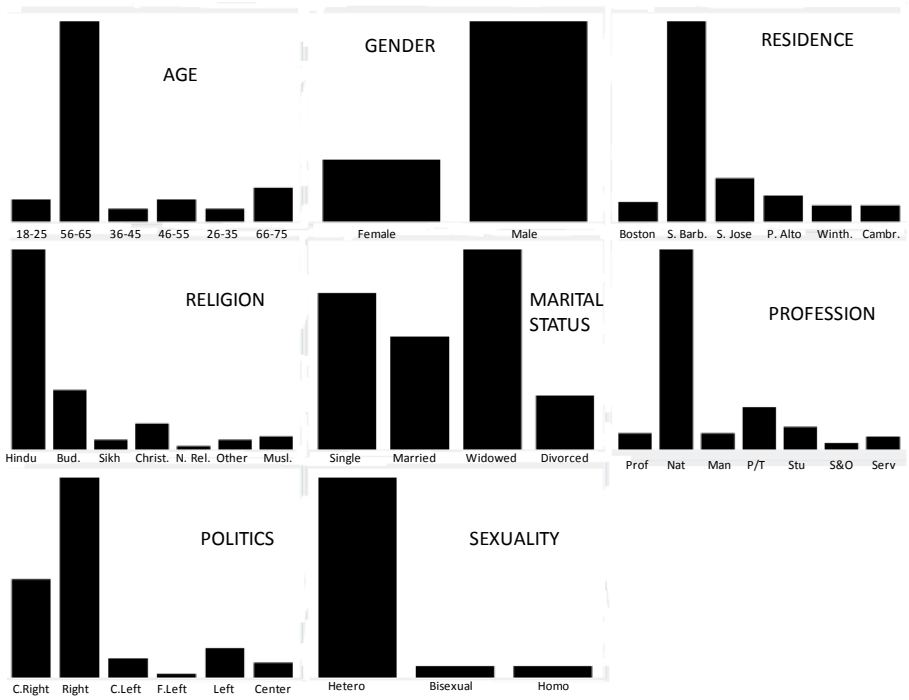

Fig. 5: Distributions of attribute-values for Community 3 (which was assigned Profile 4)

Table 7. Distributions for top 3 most frequent attribute-values for 'age', 'religion' and 'profession' (by community/profile).

Community	Assigned profile	age	religion	profession	Nº instances in community
0	0	36-45, 26-35, 46-55† {0.66, 0.19, 0.15}‡	No Rel, Muslim, Christian {0.54, 0.31, 0.15}	Sales & Off, Prof, Student {0.61, 0.19, 0.20}	216
1	2	18-25, 26-35, 36-45 {0.57, 0.33, 0.10}	Christian, Muslim, Jewish {0.69, 0.16, 0.16}	Student, Sales & Off, Prod/Trans {0.56, 0.21, 0.23}	172
2	1	26-35, 18-25, 46-55 {0.62, 0.20, 0.18}	Buddhist, Hindu, Christian {0.61, 0.25, 0.14}	Manager, Sales & Off, Prof {0.61, 0.21, 0.18}	211
3	4	56-65, 66-75, 46-55 {0.66, 0.19, 0.15}	Hindu, Buddhist, Christian {0.62, 0.27, 0.11}	Nat Rec, Prod/Trans, Student {0.63, 0.21, 0.16}	97
4	5	66-75, 76-85, 56-65 {0.62, 0.22, 0.16}	Christian, Jewish, Muslim {0.43, 0.44, 0.12}	Prod/Trans, Nat Rec, Student {0.60, 0.22, 0.17}	81
5	3	18-25, 26-35, 36-45 {0.55, 0.36, 0.10}	Muslim. Christian, Hindu {0.69, 0.20, 0.11}	Prof, Sales&Off, Manager {0.64, 0.20, 0.16}	157
6	7	18-25, 26-35, 46-55 {0.50, 0.38, 0.13}	Jewish, Christian, Other Rel {0.46, 0.42, 0.13}	Prof, Service, Sales&Off {0.50, 0.38, 0.13}	24
7	9	56-65, 36-45, 46-55 {0.40, 0.40, 0.20}	Hindu, Buddhist {0.80, 0.20, 0.00}	Nat Rec, Prod/Trans {0.60, 0.40, 0.00}	5
8	6	18-25, 26-35, 36-45 {0.64, 0.25, 0.11}	Christian, Jewish, No Rel {0.71, 0.18, 0.11}	Prof, Manager, Sales&Off {0.71, 0.21, 0.07}	28
9	8	56-65, 66-75, 18-25 {0.67, 0.33, 0.00}	Hindu, Buddhist, No Rel {0.44, 0.56, 0.00}	Nat Rec, Prod/Trans, Service {0.67, 0.33, 0.00}	9

†Top 3 categories; ‡{% top category, % 2nd and 3rd categories, % all other categories}

6.1.2 Supervised/non-supervised evaluation of matching of profiles to communities.

In Fig. 6 we see the decision tree induced by C4.5 using the community id as the class label. We have compacted the decision tree output by specifying the minimum number of objects as 10, due to space limitations. The overall precision of the model was 65% and the precision for individual communities was over 61% for all communities except C6 to C9 for which the model was unable to build predictive rules. We note that C6 to C9 have a small number of instances relative to the other communities and this causes a class imbalance problem of C4.5. The recall was over 60% for all communities except C6 to C9.

However, we must emphasize that we are not performing a data mining exercise to build the most precise supervised model possible. Our synthetic data generator purposely "hedges" the primary attribute-value (e.g. age="26-35") in a Profile to include a measured amount of similar/close attribute-values (e.g. age="18-25", "36-45"). Also, it introduces a measured amount of noise, that is, attribute values which are neither the "primary" one nor the "close" ones. Hence, the overall precision of 65% with the medium level dispersion is what we expect. We would expect that with the low dispersion the precision of the C4.5 model would go up and with the high dispersion it would go down. However, this is a trivial consequence of the resulting correlation of the community to the frequency count of the attribute values. We perform a more detailed evaluation of the distributions for different dispersion levels in Section 6.2.

With respect to the class imbalance problem, this is minimized for larger datasets. If we do wish to process small communities, different data mining solutions exist for class imbalance, one of which is 'boosting'.

An example of interpretation of the tree in Fig. 6 would be as follows: if religion="Christian" and residence = "Palo Alto" then community = 1 with a confidence level of 89/(89+10)=90%. If we reference Table 4, we see that community 1 was assigned Profile 2; then, if we reference Table 3 we see that Profile 2 was defined as being Christians living in Palo Alto. Another example would be: if religion="Christian"

and residence = “Winthrop” and age=”18-25” then community = 8 with a confidence level of 22/(22+12)=65%. If we reference Table 4, we see that community 8 was assigned Profile 6; again, if we reference Table 3 we see that Profile 6 was defined as being Christians living in Winthrop whose age is in the range 18-25.

```
religion = Christian
|   residence = Winthrop
|   |   age = 18-25: 8 (22.0/12.0)
|   |   age = 26-35: 5 (12.0/8.0)
|   |   age = 36-45: 0 (6.0/3.0)
|   |   age = 66-75: 0 (1.0)
|   |   age = 56-65: 4 (1.0)
|   |   age = 76-85: 4 (1.0)
|   |   age = 46-55: 8 (4.0/1.0)
|   residence = Palo Alto: 1 (89.0/10.0)
|   residence = Boston
|   |   maritalstatus = Single: 5 (11.0/7.0)
|   |   maritalstatus = Married: 0 (10.0/6.0)
|   |   maritalstatus = Divorced: 1 (4.0/2.0)
|   |   maritalstatus = Widowed: 4 (1.0)
|   residence = San Jose
|   |   profession = Professional: 1 (4.0/2.0)
|   |   profession = Student: 1 (11.0/3.0)
|   |   profession = Sales and office: 4 (0.0)
|   |   profession = Production transportation and material moving: 4 (22.0/2.0)
|   |   profession = Manager: 2 (3.0/2.0)
|   |   profession = Natural resources construction and maintenance: 4 (4.0/1.0)
|   |   profession = Service: 1 (2.0)
|   residence = Santa Barbara: 1 (18.0/8.0)
|   residence = Cambridge: 2 (25.0/15.0)
religion = Hindu
|   profession = Professional: 2 (16.0/11.0)
|   profession = Student: 1 (11.0/8.0)
|   profession = Sales and office: 0 (16.0/10.0)
|   profession = Production transportation and material moving: 3 (11.0/5.0)
|   profession = Manager: 2 (29.0/11.0)
|   profession = Natural resources construction and maintenance: 3 (51.0/8.0)
|   profession = Service: 2 (7.0/4.0)
religion = Muslim
|   profession = Professional: 5 (86.0/10.0)
|   profession = Student: 1 (15.0/9.0)
|   profession = Sales and office: 0 (44.0/15.0)
|   profession = Production transportation and material moving: 0 (9.0/6.0)
|   profession = Manager: 5 (18.0/10.0)
|   profession = Natural resources construction and maintenance: 3 (6.0/4.0)
|   profession = Service: 5 (14.0/7.0)
religion = No religious affiliation: 0 (137.0/20.0)
```

Fig. 6: C4.5 Pruned Tree. Complete dataset with community id as the classifier label (some level 1 nodes have been removed for brevity).

We also ran Kmeans on the dataset, using the community id for a class to cluster evaluation, with the number of clusters set to 10, which gave 62% correctly clustered instances. Finally, we applied the Weka attribute selection method “InfoGain” with the Ranker option to the dataset, which ranked the attributes in the following order, with respect to the community id: religion, profession, age, residence, political-orientation, gender, marital-status, sexual-orientation.

6.2 Real topologies – ground truth community datasets

Instead of using RMat to generate the topology, we will now use several real topologies which represent “ground truth” communities obtained from the SNAP online repository (https://snap.stanford.edu/data/).

In the previous section we recall that we generated the topology using RMat and assigned the Communities using the Louvain algorithm. Now we don’t need to generate the topology because it is a real one from real OSN apps (Amazon, YouTube and LiveJournal). Also, we don’t need to assign the communities because they are also real, calculate by online group membership in the corresponding apps. In this section we also benchmark the three different control parameter sets and evaluate the results. The control parameter sets correspond to three ‘dispersion’ levels for the data: level1 (low), level2 (medium) and level3 (high). We recall that in Section 6.1, we generated the data for the RMat topology using the level2 dispersion level.

Table 8. Ground Truth dataset statistics (5000 top communities)

Dataset name	Nodes	Edges	Nº users per community: Range (Avg.)	Nº communities per user: Range (Avg.)
Amazon	14771	87322	2-327 (177.51)	1-1614 (56.84)
YouTube	39841	448470	2-2217 (14.59)	1-54 (1.83)
Livejournal	84438	3043040	3-1441 (27.8)	1-20 (1.64)

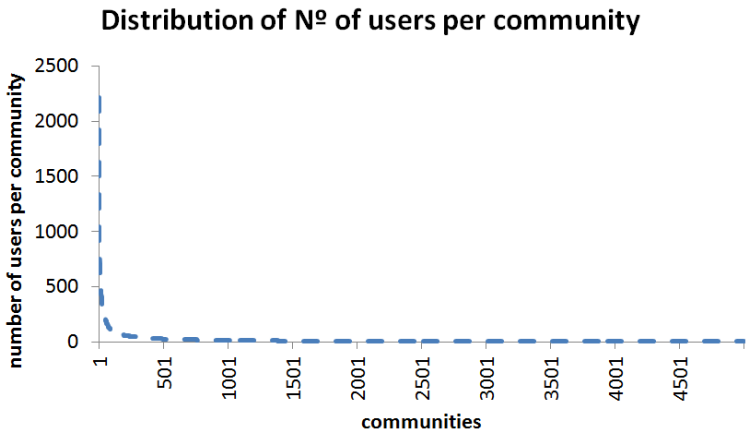

Fig. 7: Distribution of the number of users per community (Youtube dataset)

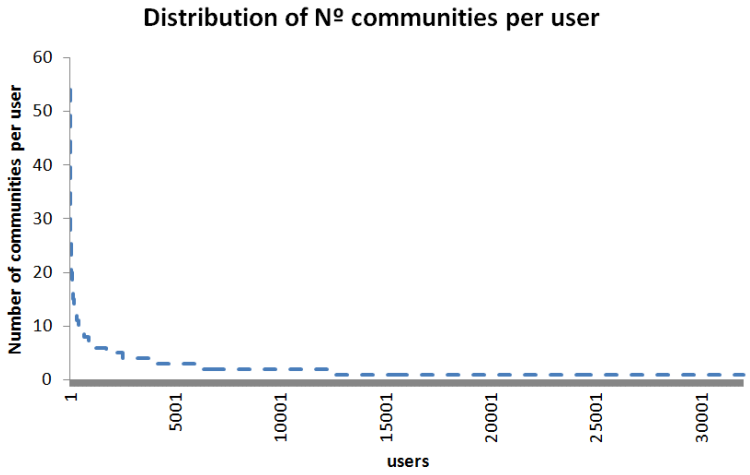

Fig. 8: Distribution of the number of communities per user (Youtube dataset)

The Amazon product co-purchasing network and ground-truth communities dataset was collected by crawling the Amazon website (Yang and Leskovec 2012). It is based on Customers Who Bought This Item Also Bought feature of the Amazon website. If a product i is frequently co-purchased with product j, the graph contains an undirected edge from i to j. Each product category provided by Amazon defines each ground-truth community.

Youtube social network and ground-truth communities (Yang and Leskovec 2012). Youtube is a video-sharing web site that includes a social network. In the Youtube social network, users form friendship each other and users can create groups which other users can join. We consider such user-defined groups as ground-truth communities. This data is provided by the reference given in (Mislove et al. 2007) and is

available at : http://socialnetworks.mpi-sws.mpg.de . In Figs. 7 and 8 we see the characteristic 'long tail' distribution of the number of communities per user and the number of users per community, respectively.

LiveJournal social network and ground-truth communities (Yang and Leskovec 2012). LiveJournal is a free on-line blogging community where users declare friendship each other. LiveJournal also allows users form a group which other members can then join. We consider such user-defined groups as ground-truth communities. We provide the LiveJournal friendship social network and ground-truth communities.

For each dataset, as described in (Yang and Leskovec 2012), each connected component in a group is considered as a separate ground-truth community. The ground-truth communities which have less than 3 nodes were removed. The datasets corresponding to the top 5,000 communities with highest quality, according to the metrics described in (Yang and Leskovec 2012) were used in the present work.

6.2.1 Data processing approach for ground truth communities

These datasets present a different scenario to the RMat synthetic dataset and communities presented in Section 6.1. Firstly, the ground truth datasets have a much higher number of communities (over 5000) whose size presents a long tail distribution and whose average size is much smaller. Secondly, a user can be a member of many communities. This scenario requires a rethink of the seed assignment, profile assignment and data propagation which we described in Section 4.

As before, we first try to assign a maximum number of seeds in each community. Then we assign the profiles to the seeds. We recall that a profile is a set of attribute-values such as those defined in Table 3.

In contrast to previously, when we had the same number of profiles to assign as communities, now we will have a fixed number of profiles (those shown in Table 3) and will assign a profile to each community based on a probability distribution. That is, each profile has a probability of being assigned between 0 and 1. Each community will have one profile assigned. However, we must take into account that a node which is a seed may also be present in many other communities. In some communities it may also be a seed and in others, not. Hence, once a node (seed, in this case) is assigned a profile it will have the same profile in all communities it is present.

Once the seeds are assigned, for each community we propagate, as before, from the seed to its immediate neighbors. Finally, as before, we assign the unassigned nodes. However, for all non-seed nodes we also have to consider they may also be present in many communities. Thus, once a node is assigned with a profile, that profile is the same for all communities in which the node is present. To facilitate this, when we assign a node for the first time, we check its community list and flag the node as assigned in all those communities. Thus the community assignment is initially lineal but then propagates out into common communities.

6.2.2 Results for ground truth communities

The results are presented for each dataset. For each dataset we show the overall statistics for each attribute in terms of top modal values and dispersion. This enables us to compare the generated data with the profile definitions.

In Table 9 we see the distribution statistics of the complete Amazon synthetic dataset for the three levels of dispersion as we defined previously in Section 5. We see that the relative percentages of the most frequent attribute values remains very constant for greater dispersion levels. This is the desired effect. We wish to maintain the overall proportions as defined in the profiles (Table 3) for the whole dataset. We will see later in Table 12 that the dispersion acts effectively and clearly at a community level. Returning to Table 9, we see, for example, that the percentage of individuals in the dataset who have an age of 18-25 years varies between 26% (level 3) and 29% (level 1). If we look at Table 3, we will see that profiles 2, 3, 6 and 7 have this age group value. The sum of the target proportions of these profiles is 17.2 + 15.7 + 2.8 + 2.4 = 38.1%. Also, we see that the percentage of individuals in the dataset who are Female is between 42% and 45%. Again, if we look at Table 3, we see that profiles 1, 3, 5, 6, 7 have this gender value. The sum of the target proportions of these profiles is 21.1+15.7+8.1+2.8+2.4=50.1. If we apply the same procedure to the religion attribute, we see in Table 9 that Christian is 28% to 29%, and in Table 3 the sum of the proportions of the profiles (2 and 6) which have that this religion value is 17.2+2.8=20.0%.

In terms of the overall proportions attribute values in the complete dataset, it is much easier to maintain these proportions for non overlapping communities; we recall that we are assigning profiles to communities. However, we recall that in the ground truth graphs, the number of users per community and the number of communities per user has a 'long tail' distribution. Thus, in order to obtain the best fit of profile assignments to communities in the desired proportions, we assign the communities in decreasing order of size. It may occur that the first community by size has a high overlap with other (smaller

communities) which may skew the overall proportions. However, this would represent the real structure of the graph so we could say the data assignment would be correct in reflecting this structure. Nevertheless, in spite of these difficulties, we can see that the attribute-values and the profiles themselves are quite well distributed and reasonably dimensioned with respect to the initial data definitions.

In general, from Table 9 we see that the profile which appears most frequently in proportional terms is {18-25, Male, Boston, Christian, Married, Sales&Off, Center Left, Hetero}. If we then refer to Table 3, we see that these attribute values are the same or are close to those which are assigned to the profiles with the highest target proportions.

In Tables 10 and 11, which show the Level 2 dispersion proportions for the YouTube and LiveJournal 'ground truth' graph datasets, we see similar trends emerging to those we have just commented for Table 9. In Tables 10 and 11 we only show Level 2 dispersion for brevity and because there is not a great variation in distributions between Levels, for the reasons we have also explained previously, this being a desirable overall property. The results of Tables 10 and 11, for the YouTube and LiveJournal show that the data assignment process can successfully assign data to significantly different topological graph structures, graph sizes, number of communities and community overlap.

In Table 12 we show a different scenario to that of Tables 9 to 11. In Table 12 we see the dispersion proportions for specific communities, profiles and dispersion levels. We recall that the dispersion is designed to act at a community level, because the data assignment process tries to assign profiles (those of Table 3) to individual communities. However, this process becomes more complex (and realistic) when we have a complex overlap of user assignment to communities. That is, a user can belong to many communities. We also recall that we assign the data profiles to individual users in a community, and once a user is assigned a profile, this profile is assigned for all the communities in which that user is a member. The first three rows of Table 12 show the assignment proportions for the attribute values for the Amazon dataset, for Profile 0 and for the three levels of dispersion. If we refer to the definition of Profile 0 in Table 3, we see that the top attribute values (those which have the highest proportion) assigned in Table 12 are indeed the same ones as defined for Profile 0. That is, the data assignment process has chosen Profile 0 as the one to be assigned to this community. If we now compare the dispersion statistics for Levels 1, 2 and 3 (rows 1 to 3), we see that as the dispersion level increases, in general, the top attributes' proportion decreases, and the proportions of the 2nd and 3rd categories, and all other categories, increases. This is what we mean by dispersion. We see that the control parameters are, in general, influencing the dispersion level as expected, for the attribute-values. However, we also see that there is not always a direct correlation of the change in proportion with dispersion levels 1 and 2. There are two exceptions: see Table 12, attribute 'profession' for Amazon profile1 and YouTube profile2. As mentioned previously, for the overlapping communities another process is acting: individuals who are members of many communities may have a greater influence on the attribute-value assignment in a community, as this may bias the proportions in a given community. This is the reason there is not always perfect correlation with the dispersion level. And this reflects the realistic "ground truth" overlapping graph structure within which we assigning the data. There is also the difficulty in matching communities in graph datasets each one of which is generated by different levels (different control parameter sets). In practice this was performed by ordering the dataset by community id and key attributes, and performing a manual inspection of the records.

However, taking into account these difficulties, the results of Table 12 do in general show a significant dispersion change between levels 1, 2 and 3. For example, for Amazon (profile 0) attribute-value 'age=35-45' has a proportion of 79% for Level1 (the least disperse), a proportion of 74% for Level2 and a proportion of 40% for Level3 (the most disperse). Likewise, for YouTube (profile 2), attribute-value 'residence=Palo Alto' has a proportion of 83% for Level1, a proportion of 50% for Level2, and a proportion of 35% for Level3.

Table 9. Amazon – complete dataset

Level of dispersion	age	gender	residence	religion	marital	profession	politics	sexuality
Level1	18-25, 26-35, 36-45† {0.29, 0.40, 0.31} ‡	Male, Female {0.54, 0.46}	Boston, Palo Alto, San Jose {0.20, 0.38, 0.42}	Christian, No Rel, Buddhist {0.29, 0.28, 0.43}	Married, Single, Divorced {0.39, 0.51, 0.10}	Sales & Off, Student, Prod/Trans {0.20, 0.35, 0.46}	Center Left, Left, Center, {0.22, 0.41, 0.37}	Hetero, Bisex, Homo {0.84, 0.14, 0.02}
Level2	18-25, 26-35, 36-45 {0.27, 0.40, 0.32}	Male, Female {0.58, 0.42}	Boston, Palo Alto, San Jose {0.22, 0.34, 0.44}	Christian, No Rel, Hindu {0.28, 0.32, 0.41}	Married, Single, Divorced {0.39, 0.50, 0.11}	Sales & Off, Student, Prof {0.22, 0.33, 0.45}	Center, Left, Center Left, {0.22, 0.42, 0.36}	Hetero, Bisex, Homo {0.81, 0.16, 0.03}
Level3	18-25, 26-35, 36-45 {0.26, 0.41, 0.33}	Male, Female {0.55, 0.45}	Palo Alto, Boston, Cambridge {0.20, 0.36, 0.44}	Christian, Buddhist, Hindu {0.29, 0.31, 0.40}	Married, Single, Divorced {0.36, 0.51, 0.13}	Sales & Off, Student, Manager {0.20, 0.36, 0.45}	Left, Center Left, Center, {0.24, 0.40, 0.36}	Hetero, Bisex, Homo {0.73, 0.23, 0.04}

†Top 3 categories; ‡{% top category, % 2nd and 3rd categories, % all other categories}

Table 10. YouTube – complete dataset

Level of dispersion	age	gender	residence	religion	marital	profession	politics	sexuality
Level2	18-25, 26-35, 36-45† {0.28, 0.42, 0.30} ‡	Male, Female {0.55, 0.45}	Palo Alto, Boston, Cambridge {0.19, 0.35, 0.46}	Christian, Muslim, No Rel {0.28, 0.30, 0.42}	Married, Single, Divorced {0.37, 0.53, 0.10}	Student, Sales & Off, Prof {0.20, 0.35, 0.45}	Left, Center Left, Center, {0.24, 0.40, 0.36}	Hetero, Bisex, Homo {0.79, 0.18, 0.03}

†Top 3 categories; ‡{% top category, % 2nd and 3rd categories, % all other categories}

Table 11. LiveJournal – complete dataset

Level of dispersion	age	gender	residence	religion	marital	profession	politics	sexuality
Level2	18-25, 26-35, 36-45† {0.28, 0.41, 0.31} ‡	Male, Female {0.54, 0.46}	Boston, Palo Alto, Winthrop {0.19, 0.35, 0.46}	Christian, Muslim, No Rel {0.30, 0.29, 0.41}	Married, Single, Divorced {0.40, 0.50, 0.10}	Sales & Off, Student, Prof {0.20, 0.34, 0.46}	Left, Center Left, Center, {0.24, 0.39, 0.37}	Hetero, Bisex, Homo {0.82, 0.15, 0.03}

†Top 3 categories; ‡{% top category, % 2nd and 3rd categories, % all other categories}

Table 12. All datasets – Dispersion levels 1 to 3 – selected individual communities

Level of dispersion	age	gender	residence	religion	marital	profession	politics	sexuality
Amazon (profile 0) Level 1	36-45, 26-35, 18-25† {0.79, 0.17, 0.05} ‡	Male, Female {0.91, 0.09}	Boston, Winthrop, Palo Alto, {0.76, 0.10, 0.14}	No Rel, Christian, Hindu {0.77, 0.16, 0.07}	Married, Single, Divorced {0.90, 0.10, 0.00}	Sales & Off, Prof, Service {0.72, 0.16, 0.12}	Center, Left, Center Left {0.72, 0.16, 0.12}	Hetero, Bisex, Homo {0.93, 0.07, 0.00}
Amazon (profile 0) Level 2	36-45, 26-35, 46-55 {0.74, 0.19, 0.07	Male, Female {0.76, 0.24}	Boston, Santa B., Palo Alto, {0.72, 0.16, 0.12}	No Rel, Christian, Hindu {0.74, 0.19, 0.07}	Married, Single, Widowed {0.77, 0.21, 0.02}	Sales & Off, Student, Manager {0.69, 0.14, 0.17}	Center, Left, Center Left {0.67, 0.21, 0.12}	Hetero, Bisex, Homo {0.88, 0.12, 0.00}
Amazon (profile 0) Level 3	36-45, 66-75, 56-65 {0.40, 0.48, 0.12}	Male, Female {0.56, 0.44}	Boston, San Jose, Winthrop {0.40, 0.40, 0.21}	No Rel, Jewish, Christian {0.35, 0.48, 0.17}	Married, Single, Widowed {0.63, 0.29, 0.08}	Sales & Off, Prod/Trans, Prof {0.38, 0.40, 0.23}	Center, Left, Far Left {0.48, 0.38, 0.15}	Hetero, Homo, Bisex {0.79, 0.15, 0.06}
Amazon (profile 1) Level 1	26-35, 18-25, 36-45 {0.75, 0.19, 0.06}	Male, Female {0.09, 0.91}	Cambridge, Palo Alto, Boston {0.75, 0.19, 0.06}	Buddhist, Hindu Christian {0.63, 0.38, 0.00}	Divorced, Married, Single {0.50, 0.50, 0.00}	Manager, Prod/Trans, Sales&Off {0.55, 0.40, 0.05}	Left, Center Left, Far Left {0.75, 0.25, 0.00}	Bisex, Hetero {0.57, 0.43, 0.00}
Amazon (profile 1) Level 2	26-35, 18-25, 56-65 {0.58, 0.22, 0.20}	Male, Female {0.12, 0.88}	Cambridge, Santa B., San Jose {0.60, 0.31, 0.10}	Buddhist, Muslim, Christian {0.60, 0.36, 0.05}	Divorced, Single, Married, {0.50, 0.44, 0.06}	Manager, Service, Sales&Off {0.63, 0.31, 0.06}	Left, Center Left, Center {0.62, 0.27, 0.11}	Hetero, Bisex, Asexual {0.53, 0.47, 0.00}
Amazon (profile 1) Level 3	26-35, 66-75, 18-25 {0.55, 0.40, 0.05}	Male, Female {0.25, 0.75}	Cambridge, Palo Alto, Boston {0.49, 0.29, 0.22}	Buddhist, Christian, Jewish {0.58, 0.29, 0.13}	Divorced, Married, Single {0.49, 0.47, 0.04}	Manager, Prod/Trans, Nat Rec {0.53, 0.33, 0.13}	Left, Far Left, Center Left {0.55, 0.43, 0.02}	Bisex, Hetero, Homo {0.44, 0.50, 0.06}
YouTube (profile 2) Level 1	18-25 {1.00, 0.00, 0.00}	Male, Female {0.93, 0.07}	Palo Alto, Cambridge {0.83, 0.17, 0.00}	Christian, Muslim {0.83, 0.17, 0.00}	Single {1.00, 0.00, 0.00}	Student, Manager {0.83, 0.17, 0.00}	Center Left, Far Left {0.83, 0.17, 0.00}	Hetero, Homo {0.83, 0.47, 0.00}
YouTube (profile 2) Level 2	18-25, 26-35, 66-75 {0.64, 0.36, 0.00}	Male, Female {0.83, 0.17}	Palo Alto, Cambridge, Santa B. {0.50, 0.43, 0.07}	Christian, Buddhist, Muslim {0.57, 0.29, 0.14}	Single, Married, Widowed {0.62, 0.32, 0.06}	Student, Manager, Prof {0.57, 0.29, 0.14}	Center Left, Right, Center {0.57, 0.36, 0.07}	Hetero, Homo {0.79, 0.21, 0.00}
YouTube (profile 2) Level 3	18-25, 26-35, 36-45 {0.30, 0.49, 0.22}	Male, Female {0.41, 0.59}	Palo Alto, Boston, Santa B. {0.35, 0.41, 0.24}	Christian, Buddhist, No Rel {0.46, 0.46, 0.08}	Single, Married, Widowed {0.57, 0.43, 0.00}	Student, Sales&Off, Prof {0.59, 0.22, 0.19}	Left, Center Left, Center {0.43, 0.49, 0.08}	Hetero, Homo {0.62, 0.38, 0.00}

†Top 3 categories; ‡{% top category, % 2nd and 3rd categories, % all other categories}

6.3 Discussion

This work presents a series of challenges which we will now comment: (i) A high degree node chosen as a seed may have a disproportionate influence on the network. This is mitigated by the use of medoid values based on the centrality metric or degree; (ii) The restrictions on the placement of the seed nodes and the topology of the communities may cause a significant percentage of nodes to have random assignments (coverage); (iii) It is easy to make non overlapping communities representative of key attribute-value profiles but we also need diversity on different attribute-values within communities and realistic overlap between communities; (iv) Obtaining 'ground truth' not just for the topology but also for the data assignments.

With respect to the attribute set chosen in the current work, some demographic attributes and their categories are standard, such as age, gender, religion, marital status, sexuality. On the other hand, attributes such as profession and especially residence will probably require customization. Application specific and activity related information, such as 'likes' and edge weights, can also be adapted by the user of the data simulator. However, changing the attribute values or introducing new attributes will require the adaption of existing control parameter rules or the creation of new rules. As we saw in Section 5, each attribute and attribute-value set requires its own customized rule.

In order to make the results comparable in the benchmarking, we have used the same set of attribute-values for all topologies. In future work we could experiment with specific attribute-values for each ground truth topology. For example, a classification of videos in the case of Youtube, blog keywords for LiveJournal or purchased product categories in the case of Amazon. In fact, one of the future benefits of our approach will be the ability to populate any published ground truth topology with customized data. With respect to data validation, one approach could be the sampling of real OSN data from these applications, in order compare the attribute-value distributions and intra-community correlations with the synthetically generated data. Indeed, this approach could be used to 'fine tune' the synthetic data for a given 'ground truth' topology.

We have seen that the high number of overlapping communities in the ground truth datasets presents a big challenge for the data assignment. We recall that in Section 6.1 we had non-overlapping communities which were significantly simpler to process than the overlapping ones of Section 6.2, and the resulting profile to community assignments fitted better because there was no inter-community interference. That is, although we assign a profile N to the seed nodes in a community A, some nodes which are not seeds in community A may be seeds in another community B, assigned with profile N'. Hence, the predominance of profile N in community A may be indirectly challenged by profile N'. As future work we will evaluate this situation in more detail. However, we propose this gives us realistic data because it is representing the overlap of the ground truth communities, and the communities will vary from being more homogeneous to more heterogeneous in nature.

7. Conclusions

In this work we have tackled the problem of generating realistic synthetic graph data which approximates an online social network, by populating a graph topology. We have tried two main approaches, the first using non-overlapping synthetic communities and the second using real overlapping 'ground truth' communities. The three step data propagation process has proved effective: seed assignment, seed neighbor assignment and assignment of the remaining nodes. As expected, the non-overlapping communities result in a more clean assignment of the profiles to the communities, whereas the overlapping communities tend to increase the 'chaos' in the data assignment process.

Using a comprehensive set of data generation parameters, we have been able to control the process which enables us to obtain a good approximation of the desired profiles, proportions, and community assignments. We can also augment the level of 'noise' in the system in terms of 'dispersion' levels defined by different control parameter sets.

We intend to put the Java source code online so that the data analysis research community can benefit from this system and adapt it to their needs.

Acknowledgements

This work is partially funded by the Spanish MEC (project TIN2013-49814-EXP). The author is grateful for the suggestions of Prof. Vladimir Estivill-Castro of the Pompeu Fabra University, Barcelona, Spain, and of Dr. Julián Salas of the University Rovira i Virgili, Tarragona, Spain.

References

Ali AM, Alvari H, Hajibagheri A, Lakkaraj K, Sukthankar G (2014) Synthetic Generators for Cloning Social Network Data. Proc. SocInfo 2014.

Ali, AM (2014) Synthetic Generators for Simulating Social Networks, 2014, Masters thesis, Univ. Florida.

Barrett CL, Beckman RJ, Khan, M., Kumar, VSA, Marathe, MV, Stretz, PE, Dutta, T, Lewis, B (2009) Generation and Analysis of Large Synthetic Social Contact Networks, Proceedings of the 2009 Winter Simulation Conference, 13-16 Dec. 2009, pp.1003 – 1014.

Blondel VD, Guillaume JL, Lambiotte R, Lefebure E (2008) Fast unfolding of communities in large networks, in Journal of Statistical Mechanics: Theory and Experiment (10), 2008, pp. 1000.

Boncz P, Perez M, Gavalda R., Angles R, Erling O, Gubichev A, Spasić M, Pham MD, Martínez N (2014) Benchmark Design for Navigational Pattern Matching Benchmarking. LDBC Cooperative Project FP7 – 317548. Coordinators: Arnau Prat , Alex Averbuch. Issue 3 28/09/2014

Chakrabarti D, Zhan Y, Faloutsos, C (2004) R-mat: A recursive model for graph mining, in Proc. SIAM Data Mining Conference, 2004. SIAM, Philadelphia, PA.

Dunbar RIM (1993) Coevolution of neocortical size, group size and language in humans. Behavioral and Brain Sciences, 16,4 (1993) 681–735.

EU's Data Protection Directive (2015) - Justice, Protection of personal data. Available at: http://ec.europa.eu/justice/data-protection/ .

Jones, R., Kumar, R., Pang, B. and Tomkins, A. (2007). "I know what you did last summer": Query logs and user privacy, Sixteenth ACM Conf. on Information and Knowledge Management, ser. CIKM. 2007, pp. 909–914.

Kelly, H. (2012) "83 million Facebook accounts are fakes and dupes". CNN, August 3, 2012. Available at: http://edition.cnn.com/2012/08/02/tech/social-media/facebook-fake-accounts/

Kossinets G, Watts D. (2006). Empirical analysis of an evolving social network, Science 311 (5757) (2006) 88–90.

Leskovec J, Kleinberg J, Faloutsos C (2005) Graphs over time: densification laws, shrinking diameters and possible explanations, in Proc. KDD '05, 11th ACM SIGKDD Int. Conf. of Knowledge Discovery and Data Mining, 2005, pp. 177-187.

McAfee, A., Brynjolfsson, E. (2012) Big Data: The Management Revolution, Harvard Business Review, October 2012 Issue.

Mislove A, Marcon M, Gummad, KP, Druschel P, Bhattacharjee B (2007) Measurement and Analysis of Online Social Networks, Proc. IMC '07, 7th ACM SIGCOMM Conference on Internet Measurement, pp. 29-42.

Nettleton, DF (2015) Generating synthetic online social network graph data and topologies, 3rd Workshop on Graph-based Technologies and Applications (Graph-TA), UPC, Barcelona, Spain, March 18th 2015.

Nettleton DF, Salas J (2014) A Data Driven Anonymization Method for Information Rich OSN Graphs. Submitted to the Journal "Expert Systems with Applications", Dec. 2014.

Nettleton DF (2013) Data mining of social networks represented as graphs, Computer Science Review 7, 1-34 (2013).

Pérez-Rosés H, Sebé F (2014) Synthetic Generation of Social Network Data With Endorsements, Journal of Simulation, (14 November 2014) | doi:10.1057/jos.2014.29

Pham, MD, Boncz P, Erling O (2013) S3G2: a Scalable Structure-correlated Social Graph Generator, Selected Topics in Performance Evaluation and Benchmarking, LNCS Vol. 7755, 2013, pp 156-172

Ramakrishnan N, Keller B, Mirza BJ. (2001). A. Grama, and G. Karypis, "Privacy risks in recommender systems," IEEE Internet Computing, vol. 5, no. 6, pp. 54–62, 2001.

Robins G, Pattison, P, Woolcock, J (2005) Small and Other Worlds: Global Network Structures from Local Processes. American Journal of Sociology (AJS), Volume 110, Number 4 (2005), 894-936.

Sala A, Cao L, Wilson C, Zablit R, Zheng H, Zhao BY (2010) Measurement-calibrated Graph Models for Social Network Experiments, WWW 2010, April 26–30, 2010, Raleigh, North Carolina, USA.

Tang L, Liu H, Zhang J, Nazeri N (2008). Community evolution in dynamic multi-mode networks, in: Proc. of the 14th ACM SIGKDD, KDD '08, New York, NY, USA, 2008, pp. 677–685

Viswanath, B, Mislove A, Cha M, Gummadi, KP. (2009). On the Evolution of User Interaction in Facebook, Proc. 2nd ACM workshop on Online Social Networks, WOSN'09, Barcelona, Spain, 2009, pp. 37–42.

Weil, J. (2015) "Mark Zuckerberg: Creator of Facebook", Abdo Publishing, Minneapolis, USA. Ed. Arnold Ringstad, ISBN 978-1-62403-647-7 (2015).

Yang J, Leskovec J (2012) Defining and Evaluating Network Communities based on Ground-truth. ICDM, 2012.

Appendix 1 –Pseudo code of synthetic data generator

Procedure Synthetic_Data_Generator_1 // non overlapping communities
Input: Number V of vertices and E of edges,
control parameter set $\mathbb{C}P$ = [NS, {SP}, PR, {DT}, RV] // see Section 5 for definitions
Output: graph G populated with data
1. *RMat*
2. **For** |V| vertices and |E| edges **generate** an OSN-like topology.
3. *Communities*
4. **Calculate** communities using Leuven method and assign community tag to each vertex.
5. *Calculate* medoid values MC for each community using centrality metric
6. Calculate distance Cdn of each node n in each community to mediod MC
7. Order nodes in each community by distance Cdn
8. S = **Seed_Assigner**(G, nSeeds)
9. // *Assign data to seeds and their neighbors in each community*
10. **For each** community c ∈ C do
11. S_c is the set of seed nodes in community c
12. NS_c **= Assign_Data_to_Seeds_and_Neighbor_Vertices_in_Community (**S_c, c, $\mathbb{C}P$**)**
13. **//** NS_c is the set of seeds and neighbors in c with data assigned
14. // *Assign data to remaining nodes in each community*
15. **Vc = Assign_Unassigned_Vertices_in_Community**(NS_c, c, $\mathbb{C}P$)
16. // Vc is the set of all nodes in c with data assigned
17. **End do** // for each community
18. **For each** edge e connected to n, n' in G **do**
19. Assign a weight between 0 and 1 based on calculated distance between respective attribute-values.
20. **End Procedure**

Procedure Synthetic_Data_Generator_2 // overlapping ground truth communities
Input: Graph topology G, control parameter set $\mathbb{C}P$ = [NS, {SP}, PR, {DT}, RV] // see Section 5 for definitions
Output: graph G populated with data
1. *Read ground truth community graph*
2. *Communities*
3. **Assign** communities from ground truth labels by assigning community tag to each vertex.
4. *Calculate* medoid values MC for each community using centrality metric
5. Calculate distance Cdn of each node n in each community to mediod MC
6. Order nodes in each community by distance Cdn
7. S = **Seed_Assigner**(G, nSeeds)
8. *Assign data to seeds and their neighbors in each community*
9. **For each** community c ∈ C do
10. S_c is the set of seed nodes in community c
11. NS_c **= Assign_Data_to_Seeds_and_Neighbor_Vertices_in_Community (**S_c, c, $\mathbb{C}P$**)**
12. **//** NS_c is the set of seeds and neighbors in c with data assigned
13. // *Assign data to remaining nodes in each community*
14. **Vc = Assign_Unassigned_Vertices_in_Community**(NS_c, c, $\mathbb{C}P$)
15. // Vc is the set of all nodes in c with data assigned
16. **End do** // for each community
17. **For each** edge e connected to n, n' in G **do**
18. Assign a weight between 0 and 1 based on calculated distance between respective attribute-values.
19. **End Procedure**

Procedure Seed_Assigner
Input: graph G, number of seeds desired nSeeds
Output: seed set S
1. **While** number of seeds assigned less than **nSeeds or** max iterations exceeded **do**
2. **For each** community c ∈ C in G **do**
3. **While** more seeds assignable **do**
4. **Choose** a vertex w from the set of nodes ordered by their centrality metric such that:
Each s ∈ S is at least at distance 2 from w.
End do
5. **End do**
6. **Save best configuration S' so far**
7. **End do**
8. **End Procedure**

Procedure Assign_Data_to_Seeds_and_Neighbor_Vertices_in_Community
Input: S_c , the set of seeds in c; c, the current community id; control parameter set $\mathbb{C}P$
Output: NS_c , set of seeds and neighbor vertices with data assigned in community c
1. **For each** vertex s ∈ S_c **do**
2. **Assign** corresponding profile p_c to attributes of s
3. Let Nv_c be the set of neighbors of s_c
4. **For each** n ∈ Nv_c **do**
5. **For each** attribute a of n **do**
6. **For each** value v of attribute n **do**
7. Assign {a, v} of ad_c to neighbor *n* according to $\mathbb{C}P$
8. **End do**
9. **End do**
10. **End do**
11. **End do**
12. **End Procedure**

Procedure Assign_Unassigned_Vertices_in_Community
Input: NS_c , the set of vertices in c with data assigned; c, the current community id; control parameter set $\mathbb{C}P$
Output: assigned set of vertices Vc in community c
1. **For each** n in c ∉ NA_c **do**
2. **For each** attribute a of n **do**
3. **For each** value v of attribute n **do**
4. **Calculate** average or modal value of
corresponding attribute-value of neighbors
of n as {n', a', v'}
5. Assign {a', v'} or random value *{a'', v''}* to *n according to* $\mathbb{C}P$
6. **End do**
7. **End do**
8. **End Procedure**