Network Analysis Software

This article points the reader to main repositories of listings of network analysis software, provides strategic guidance in selecting the appropriate tool or set of different tools for a network analysis project, and highlights trends and needs with respect to network analysis software. A collection of pointers to all tools and web pages mentioned in this article as well as to selected additional learning resources is located at the end of this article. Social networking services are not included in this reference list.

Network analysis software is used to analyze relational data, and typically provides statistical routines and visualization mechanisms. A plethora of network analysis software tools is available, and new products are frequently released. This trend has greatly contributed to the advancement, power and flexibility of network science, but also confronts anybody interested in network analysis, especially newcomers, with an overcrowded pool of choices between products. As network analysis keeps being leveraged into new domains and morphing out of established fields such as sociology, anthropology, and physics, methodological variations and innovative computational solutions are being added to the network analysis toolbox. The increasing popularity of network analysis, which goes hand in hand with the widespread adoption of social networking sites and applications, has created a need for analytical solutions that people from all backgrounds can learn how to use; moving network analysis into the realm of general utility methods such as statistics. One out of many examples for this development is *NodeXL*, a network analysis plug-in for Microsoft Excel that was released in July 2008. NodeXL does not require a steep learning curve, easily integrates with main stream software, and features the collection of relational data from popular social networking sites such as *Flickr*, *Twitter*, and *YouTube* as well as from personal email repositories.

Information Resources for Network Analysis Software

Various repositories and reviews provide descriptions of subsets of network analysis software products that were selected and detailed along a small number of more or less explicitly stated criteria and dimensions:

The International Network for Social Network Analysis (INSNA) offers a publicly available listing of currently 23 network analysis tools provided by its members. INSNA also kept the predecessor of the current tools page online, which provides a brief description and a link to over 60 tools.

In 2005, Mark Huisman and Marijtje van Duijn provided a review of 23 stand-alone packages and five software libraries. They deliver a detailed description and comparison of the functionalities, documentation, and user-friendliness of six widely used network analysis tools, namely *MultiNet*, *NetMiner*, *Pajek*, *StOCNET*, *STRUCTURE*, and *UCINET*.

Given the large and growing number of network analysis software, no single person or small group of people might want to carry the burden of building and maintaining a systematic and comprehensive overview of these tools. However, if carried out collectively, this goal might be achievable with very little individual effort. Leveraging on the idea of the wisdom of crowds, a few network researchers started a *Wikipedia page for social network analysis software* in early 2008 to serve as a neutral point of access outside of specific research fields and professional organizations. This page specifies the name, URL, main functionality, supported data formats and platforms, license type, cost, and additional notes for currently over 40 tools. Since this page can be edited by anyone, some tools might not be related to network analysis as it is understood in this book. However, the page gives an up to date overview that is consistently improved by the community.

Software for the Network Analytical Research Process

Ultimately, any network analysis software serves to facilitate one or more of the steps involved in the overall network analysis process. The next section outlines this process and gives some advice on how to select the appropriate tools for putting this process into practice.

The network analysis process is less standardized than research methodologies in other fields. This fact can be partially attributed to the wide diffusion of network analysis across fields as diverse as anthropology, physics, and economics. As methodological variations keep being added and dropped, a patchwork landscape of steps and subroutines has emerged. The confinement of steps has recuperations on a review of social network analysis software. For example, some researchers and practitioners alike might only consider the actual relational data analysis step as the network analysis method. Synthesizing various descriptions of formal network analysis methodology leads to a more comprehensive approach. This article does not offer a unified procedural framework for conducting network-oriented studies, but suggests that at a minimum, a network analysis project should include the following steps:

1. Specify a goal, task, or research question. This objective can, for example, be posed by a client, emerge from a systematic review of prior work in the form of a gap or a contradiction in findings, or serve to explore an uninvestigated phenomenon.
2. Specify the entities (nodes), relations (edges) between these entities, and network boundaries that are relevant and appropriate for the given project.
3. If no data is given, collect relational data.
4. Represent the relational data. Typically, this is done if the form of lists, matrices, or other types of structured data, such as XML (extensible markup language).
5. Analyze and utilize relational data. This may entail database operations such as storage, search, and retrieval, network analytical operations such as computing network analytical measures, network visualization, network simulation, and generating input for machine learning systems, among other options.
6. Validate the results. Perform error analysis if applicable.
7. Interpret the results with respect to step 1. This can include the suggestion of intervening strategies and policies.

Steps three to seven lend themselves to computer support. The degree of automation can range from computerized support, e.g. for creating and drawing structural models that need to be tested or for designing and disseminating surveys, to full automation, e.g. for computing network analytical measures and producing graph visualizations. However, there is no single Swiss Army knife-like software product that supports all of these steps. For example, network data collection tools such as *Network Genie* or *SurveyMonkey* rarely exhibit network analytical capabilities so that the users of these tools need to adopt the given features in creative ways. It rather is the case that most network analysis tools are designed to support a specific subset of possible functionalities. The Wikipedia page devoted to social network analysis software provides some quantitative information on this issue. Based on this limited sample, the following main categories of functionalities and associated ratio of tools can be identified: out of 42 tools considered, 88% compute network analytical measures, 69% feature network visualization, 19% support network data collection, and 10% are capable of simulating networks. No single tool combines all of these functionalities.

What does the modularization of functionalities across different tools mean for people who are planning to conduct a network study? First, the availability of multiple tools for different network analytical procedures equips users with great flexibility and has stimulated a competitive, vivid and innovative tool production environment. Second, it is highly likely that one will need to assemble a tool chain that supports the functionalities needed for the project at hand instead of searching for an all-in-one solution. The composed tool chain might vary across projects and datasets.

The data from the Wikipedia page suggests some additional trends: first, an increasing number of tools support the handling of time series, temporal, and spatial data, for example *ORA*. Second, tools customized for certain domains have started to emerge, for instance for the financial sector, e.g. *Detica NetReveal* and *Financial Network Analyzer*, for ecology, e.g. *EvESim*, and for communication networks, e.g. *Idiro SNA Plus*. Third, 43% of the tools are available as open source software, 38% require the purchase of a license, and 19% can be used for free for research and for a fee for commercial purposes.

In order to make an informed and comprehensive selection of the appropriate mix-and-match of tools, users should consider some critical features of network analysis software when it comes to designing a network study: interoperability, scalability, and customizability. The remainder of this article explains each of these features in more detail.

Interoperability

Network analysis tools do not use a single shared data format for input and output. Network data formats vary along multiple dimensions: first, some tools use proprietary formats, such as UCINET, while most tools work with openly available and specified formats, such as the *DyNetML* format employed by the *CASOS* tools. Second, the format might represent networks as lists, e.g. Pajek's .net, as matrices, e.g. in UCINET, or in an extensible markup language (*XML*), such as *GraphML*. Generating lists and matrices requires basic technical skills, and in fact can be accomplished by using common text and spreadsheet editors. In contrast to that, producing files in any XML derivate is not as simple. This difference might matter when one wants to be able to quickly build, test or experiment with sample files. However, from an automation perspective, XML files are convenient and efficient since their specification has all information that a machine needs for validating or parsing a file and knowing where to look for an error. In contrast to that, any non-standard syntax and semantics of data formats require a steeper learning curve and more time and effort for debugging and verifying files. Finally, not all network data formats are capable of expressing the directionality and weight of edges, attributes of nodes and/ or edges, and multiple types of nodes (multimodal data) and edges (multiplex data).

To avoid file format incompatibilities, it might be necessary to investigate the characteristics of the network data prior to selecting suitable tools. If multiple tools are to be employed, users also need to check upfront if the output from one tool can be read as input by the next tool in line. This potential problem is mitigated by the fact that nowadays, most tools feature the conversion of inputs and outputs to various common network data formats such as *.csv* – a format that can be read and written by most spreadsheet editors, and *.txt*, which can be loaded into most text editors. Furthermore, the Wikipedia page for social network analysis software specifies the accepted input and output format for most of the listed tools.

Scalability

Networks can be large, such as the Internet, and complex, such as the protein interactions in a cell. Therefore, it is necessary to identify the upper bound of the number of nodes and edges in the data to be able to start an informed search for a tool that can handle that amount. Not all packages specify the limit of their capabilities in that respect. A special case for scalability is network visualization: Displaying large numbers of nodes on a screen might neither be technically feasible nor cognitively perceivable, with the former potentially outperforming the latter. What is needed here are techniques for the reduction and abstraction of network data to the gist of the information that is relevant to the user and the project at hand. Conventional network-centric methods for this purpose are clustering and the removal of nodes and edges that occur infrequently or scale low on some other dimensions of importance. Alternatively, one could leverage on visualization approaches and respective software that do not display network data as network pictures, but yet elicit the key dynamics and driving forces of networked systems in a graphical fashion. An example is the *history flow* software product that shows the edit history of Wikipedia pages – a co-authoring network in network analysis terms.

Further aspects of the scalability of software are the automated repeatability of analyses and the capability to distribute computations across multiple time sequences and/or machines. These needs are supported by tools that offer a script-based version and are based on threaded code, respectively.

Customizability

The more a network analysis software is customizable, the more effort is needed in making the tool work. Closed source tools do not allow the user to go beyond the capabilities made available in a given release; thus providing no flexibility in the sense of customizability. Exceptions to this rule are tools that are equipped with an application programming interface (API). Open source tools allow the user to modify a product and extending it. Open source libraries provide readily implemented functionalities that can be integrated into existing or new tools. Open source software allows for the greatest flexibility, but requires respective technical skills. Examples for widely used libraries in the network analysis domain are the Java Universal Network/Graph (*JUNG*) Framework in Java, NodeXL written in C#/.Net, a set of *R* libraries, and the *Network Workbench*, which is written in different programming languages.

In the same way as network analysis tools use different data formats, social networking sites – currently a prominent source of network data – have implemented different ways of storing and allowing access to their data. An example for a software component that enables users to interact with various web-based social networking sites is the closed-source API *OpenSocial*. This API was developed by Google in cooperation with other companies and was released in 2007. OpenSocial allows users to access data and core features of participating social network applications such as *Friendster*, *Hyves*, *LinkedIn*, *MySpace*, *orkut*, and *XING*; thereby contributing to the interoperability and fusion of data and services.

References and further readings

Carley, Kathleen M., Jana Diesner, Jeffrey Reminga, and Maksim Tsvetovat. “Toward an interoperable dynamic network analysis toolkit," Decision Support Systems. Special Issue Cyberinfrastructure for Homeland Security 43 (2007): 1324-47.
Huisman, Mark Van Duijn, Marijtje "Software for social network analysis," in Models and methods in social network analysis, Carrington, Peter J., John Scott, Stanley Wasserman (eds.), New York: Cambridge University Press, 2005, 270-316.

The following pointers are current as of January 2010.
URLs for software descriptions repositories:

- INSNA Member Listed Software: http://www.insna.org/software/index.html
- Wikipedia page for social network analysis software: http://en.wikipedia.org/wiki/Social_network_analysis_software
- CASOS - Center for Computational Analysis of Social and Organizational Systems: http://www.casos.cs.cmu.edu/computational_tools/tools.html

URLs for software products and open source libraries:

- Detica NetReveal http://www.deticanetreveal.com/
- EvESim http://www.evesim.org/
- Financial Network Analyzer: http://www.financialnetworkanalysis.com/
- HistoryFlow: http://www.research.ibm.com/visual/projects/history_flow/
- Idiro SNA Plus: http://www.idiro.com/sna-plus/index.html
- JUNG: http://jung.sourceforge.net/
- MultiNet: http://www.sfu.ca/personal/archives/richards/Multinet/Pages/multinet.htm
- NetMiner: http://www.netminer.com
- Network Genie: https://secure.networkgenie.com/
- Network Workbench: http://nwb.slis.indiana.edu/nwb1.0.0official_releasenotes.html
- NodeXL: http://nodexl.codeplex.com/
- OpenSocial: http://wiki.opensocial.org
- ORA: http://www.casos.cs.cmu.edu/projects/ora/
- Pajek: http://pajek.imfm.si/doku.php
- R: http://cran.r-project.org/web/packages/sna/index.html
- StOCNET: http://stat.gamma.rug.nl/stocnet/
- STRUCTURE: http://faculty.chicagobooth.edu/ronald.burt/teaching/
- SurveyMonkey: http://www.surveymonkey.com/
- UCINET: http://www.analytictech.com/ucinet/

URLs for network data formats:

- DyNetML: http://www.casos.cs.cmu.edu/projects/dynetml/
- GraphML: http://graphml.graphdrawing.org/index.html
- Overview: http://mdround.blogs.com/SNA_Tools_and_Formats_20091116.pdf

SEE ALSO: Methods of Data Collection, Network Visualization, Network Simulations

Jana Diesner, Carnegie Mellon University
Ines Mergel, Syracuse University
Kathleen M. Carley, Carnegie Mellon University