Towards a format for describing networks
Format elements

Vladimir Batagelj[1,2,3,4][0000−0002−0240−9446],
Tomaž Pisanski[1,2][0000−0002−1257−5376], Iztok Savnik[1][0000−0002−3994−4805],
Ana Slavec[1][0000−0002−0171−2144], and Nino Bašić[1,2][0000−0002−6555−8668]

[1] UP FAMNIT Koper
[2] IMFM Ljubljana
[3] UL FMF Ljubljana
[4] vladimir.batagelj@fmf.uni-lj.si
[5] https://github.com/bavla/netsJSON

version: March 19, 2025 at 06 : 17

Abstract. The key components that a common format for describing networks should include are discussed.

Keywords: First keyword · Second keyword · Third keyword

1 Introduction

In 2023, the International Network for Social Network Analysis (INSNA) requested that Zachary Neal form a working group to develop **recommendations for sharing network data and materials**. They were published in *Network Science* in 2024 [15] accompanied with the *Endorsement page* [14].

It would be useful to have a common "archiving/intermediate" format that can describe (almost) all networks. It is easy to write converters from this format to a selected format or corresponding network reading procedures.

1.1 Software support for network analysis

There are many tools and programs for network analysis UCINET, Pajek, Gephi, NetMiner, Cytoscape, NodeXL, E-Net, Tulip, PUCK, GraphViz, SocNetV, Kumu, Polinode, etc.

Programmers can use network analysis packages/libraries in different programming languages

- **Python:** NetworkX, igraph, Snap.py, graph-tool, NetworKit, PyGraphistry, Nets, cdlib, node2vec, DGL, PyG, Tulip, PyVis,
- **R:** igraph, statnet, sna, qgraph, RSiena, tnet, multiplex, NetSim, influenceR, tidygraph, intergraph, netUtils, ggraph, networkD3, visNetwork, DiagrammeR, graphlayouts, ndtv,

- **Julia:** LightGraphs, Graphs, MetaGraphs, SimpleWeightedGraphs, Erdos, MultilayerGraphs, GraphDataFrameBridge, GraphIO, NetworkDynamics, TemporalGPs, EcologicalNetwork, CommunityDetection, GraphPlot, NetworkLayout,
- **C++:** Boost Graph Library, igraph, SNAP, NetworKit, NetworkX, Graph-tool, GraphBLAS, Lemon Graph Library, GraphHopper, Gelly, Tulip, OGDF,
- etc.

They are supporting different network description formats: CSV, UCINET DL, Pajek NET, Gephi GEXF, GDF, GML, GraphML, GraphViz DOT, Tulip TPL, Netdraw VNA, Spreadsheet, etc. [10].

In addition, network data appears in several application areas such as chemistry and genealogy. There are many formats for describing molecular graphs: Molfile, SDF, CML, PDB, XYZ, CIF, FASTA, CDX, CDXML, JCAMP-DX, SMILES, InChI, and others. The most widely used format for genealogical data exchange, GEDCOM is a plain text file format that stores information about individuals, families, events, and sources. It has several derivatives. It is considered an exchange format between various genealogy programs, which are often based on their own format. Some of the most well-known are Ancestry Tree Files, Family Tree Maker, Legacy Family Tree, RootsMagic, OpenGen Alliance, Open Archives Format, FamilySearch JSON, Gramps XML, TEI, PROGEN, Webtrees, PAF.

Tomaž Pisanski et al. – Vega

1.2 Network representations

There are three commonly used file representations of graphs and networks.

- **Link list (with weights)** This is the most commonly used and expressively most flexible representation.
- **Matrix representation** It is often found in older sources. It is suitable for describing smaller, denser simple networks. We lose the distinction between directed and undirected and multiple links. For larger networks, which are usually sparse, it requires a lot of space – most matrix entries have the value 0.
- **Neighbor sets** This representation is very economical but only useful for networks without link properties.

1.3 Repositories of graph and network data

- ICON – Colorado Index of Complex Networks
- UCINET datasets
- Pajek networks
- UCI Network Data Repository
- CASOS – **C**omputational **A**nalysis of **S**ocial and **O**rganizational **S**ystems
- SNAP – **S**tanford **N**etwork **A**nalysis **P**latform

- KONECT – **Ko**blenz **Ne**twork **C**ollec**t**ion
- Netzschleuder – network catalogue, repository and centrifuge
- Schochastics network data
- Network Repository
- Siena data sets
- The House of Graphs
- Encyclopedia of Graphs
- Datasets of Highly Symmetric Objects
- Awesome Network Analysis datasets
- Kinsources
- RCSB Protein Data Bank

2 Description of traditional networks

2.1 Description of networks using a spreadsheet

How to describe a network $\mathcal{N} = (\mathcal{V}, \mathcal{L}, \mathcal{P}, \mathcal{W})$? In principle the answer is simple – we list its components $\mathcal{V}$, $\mathcal{L}$, $\mathcal{P}$, and $\mathcal{W}$. The simplest way is to describe a network $\mathcal{N}$ by providing $(\mathcal{V}, \mathcal{P})$ and $(\mathcal{L}, \mathcal{W})$ in a form of two tables.

As an example, let us describe a part of the network determined by the bibliographical data about the following works: Generalized blockmodeling, Clustering with relational constraint, Partitioning signed social networks, The Strength of Weak Ties.

There are nodes of different types (modes): persons, papers, books, series, journals, publishers; and different relations among them: author_of, editor_of, contained_in, cites, published_by. For some types of nodes additional properties are known: sex, year, volume, number, first and last page, etc.

Both tables are often maintained in Excel. They can be exported as text in CSV (Comma Separated Values) format. Tables for our example are given in Figures 1 and 2. In large networks, we split a network into some subnetworks – a collection, to avoid the empty cells.

2.2 Factorization and description of large networks

To save space and improve computing efficiency we often replace values of categorical variables with integers. In R this encoding is called a *factorization*.

We enumerate all possible values of a given categorical variable (coding table) and afterward replace each value with the corresponding index in the coding table. Since node labels/IDs can be considered a categorical variable, factorization is usually applied also on them.

This approach is used in most programs dealing with large networks. Unfortunately, the coding table is often considered as a kind of meta-data and is omitted from the description.

Be careful, in data analysis, indices start with 1, but real computer scientists start counting from 0.

```
name;mode;country;sex;year;vol;num;fPage;lPage;x;y
"Batagelj, Vladimir";person;SI;m;;;;;;809.1;653.7
"Doreian, Patrick";person;US;m;;;;;;358.5;679.1
"Ferligoj, Anuška";person;SI;f;;;;;;619.5;680.7
"Granovetter, Mark";person;US;m;;;;;;145.6;660.5
"Moustaki, Irini";person;UK;f;;;;;;783.0;228.0
"Mrvar, Andrej";person;SI;m;;;;;;478.0;630.1
"Clustering with relational constraint";paper;;;1982;47;;413;426;684.1;380.1
"The Strength of Weak Ties";paper;;;1973;78;6;1360;1380;111.3;329.4
"Partitioning signed social networks";paper;;;2009;31;1;1;11;408.0;337.8
"Generalized Blockmodeling";book;;;2005;24;;1;385;533.0;445.9
"Psychometrika";journal;;;;;;;;741.8;086.1
"Social Networks";journal;;;;;;;;321.4;236.5
"The American Journal of Sociology";journal;;;;;;;;111.3;168.9
"Structural Analysis in the Social Sciences";series;;;;;;;;310.4;082.8
"Cambridge University Press";publisher;UK;;;;;;;534.3;238.2
"Springer";publisher;US;;;;;;;884.6;174.0
```

Fig. 1. File `bibNodes.csv` – $(\mathcal{V}, \mathcal{P})$ table for nodes

```
from;relation;to
"Batagelj, Vladimir";authorOf;"Generalized Blockmodeling"
"Doreian, Patrick";authorOf;"Generalized Blockmodeling"
"Ferligoj, Anuška";authorOf;"Generalized Blockmodeling"
"Batagelj, Vladimir";authorOf;"Clustering with relational constraint"
"Ferligoj, Anuška";authorOf;"Clustering with relational constraint"
"Granovetter, Mark";authorOf;"The Strength of Weak Ties"
"Granovetter, Mark";editorOf;"Structural Analysis in the Social Sciences"
"Doreian, Patrick";authorOf;"Partitioning signed social networks"
"Mrvar, Andrej";authorOf;"Partitioning signed social networks"
"Moustaki, Irini";editorOf;"Psychometrika"
"Doreian, Patrick";editorOf;"Social Networks"
"Generalized Blockmodeling";containedIn;"Structural Analysis in the Social Sciences"
"Clustering with relational constraint";containedIn;"Psychometrika"
"The Strength of Weak Ties";containedIn;"The American Journal of Sociology"
"Partitioning signed social networks";containedIn;"Social Networks"
"Partitioning signed social networks";cites;"Generalized Blockmodeling"
"Generalized Blockmodeling";cites;"Clustering with relational constraint"
"Structural Analysis in the Social Sciences";publishedBy;"Cambridge University Press"
"Psychometrika";publishedBy;"Springer"
```

Fig. 2. File `bibLinks.csv` – $(\mathcal{L}, \mathcal{W})$ table for links

```
# transforming CSV file to Pajek files, by Vladimir Batagelj, June 2016
colC <- c(rep("character",4),rep("numeric",5)); nas=c("","NA","NaN")
nodes <- read.csv2("bibNodes.csv",encoding='UTF-8',colClasses=colC,na.strings=nas)
n <- nrow(nodes); M <- factor(nodes$mode); S <- factor(nodes$sex)
mod <- levels(M); sx <- levels(S); S <- as.numeric(S); S[is.na(S)] <- 0
links <- read.csv2("bibLinks.csv",encoding='UTF-8',colClasses="character")
F <- factor(links$from,levels=nodes$name,ordered=TRUE)
T <- factor(links$to,levels=nodes$name,ordered=TRUE)
R <- factor(links$relation); rel <- levels(R)
net <- file("bib.net","w"); cat('*vertices ',n,'\n',file=net)
clu <- file("bibMode.clu","w"); sex <- file("bibSex.clu","w")
cat('%',file=clu); cat('%',file=sex)
for(i in 1:length(mod)) cat(' ',i,mod[i],file=clu)
cat('\n*vertices ',n,'\n',file=clu)
for(i in 1:length(sx)) cat(' ',i,sx[i],file=sex)
cat('\n*vertices ',n,'\n',file=sex)
for(v in 1:n) {
  cat(v,' "',nodes$name[v],'"\n',sep='',file=net);
  cat(M[v],'\n',file=clu); cat(S[v],'\n',file=sex)
}
for(r in 1:length(rel)) cat('*arcs :',r,' "',rel[r],'"\n',sep='',file=net)
cat('*arcs\n',file=net)
for(a in 1:nrow(links))
  cat(R[a],': ',F[a],' ',T[a],' 1 l "',rel[R[a]],'"\n',sep='',file=net)
close(net); close(clu); close(sex)
```

Fig. 3. `CSV2Pajek.R` – program for converting tables into network in Pajek format

```
*vertices  16
1 "Batagelj, Vladimir"
2 "Doreian, Patrick"
3 "Ferligoj, Anuška"
4 "Granovetter, Mark"
5 "Moustaki, Irini"
6 "Mrvar, Andrej"
7 "Clustering with relational constraint"
8 "The Strength of Weak Ties"
9 "Partitioning signed social networks"
10 "Generalized Blockmodeling"
11 "Psychometrika"
12 "Social Networks"
13 "The American Journal of Sociology"
14 "Structural Analysis in the Social Sciences"
15 "Cambridge University Press"
16 "Springer"
*arcs :1 "authorOf"
*arcs :2 "cites"
*arcs :3 "containedIn"
*arcs :4 "editorOf"
*arcs :5 "publishedBy"
```

```
*arcs
1: 1 10 1 l "authorOf"
1: 2 10 1 l "authorOf"
1: 3 10 1 l "authorOf"
1: 1 7 1 l "authorOf"
1: 3 7 1 l "authorOf"
1: 4 8 1 l "authorOf"
4: 4 14 1 l "editorOf"
1: 2 9 1 l "authorOf"
1: 6 9 1 l "authorOf"
4: 5 11 1 l "editorOf"
4: 2 12 1 l "editorOf"
3: 10 14 1 l "containedIn"
3: 7 11 1 l "containedIn"
3: 8 13 1 l "containedIn"
3: 9 12 1 l "containedIn"
2: 9 10 1 l "cites"
2: 10 7 1 l "cites"
5: 14 15 1 l "publishedBy"
5: 11 16 1 l "publishedBy"
```

Fig. 4. File `bib.net` – bibliography network in Pajek format

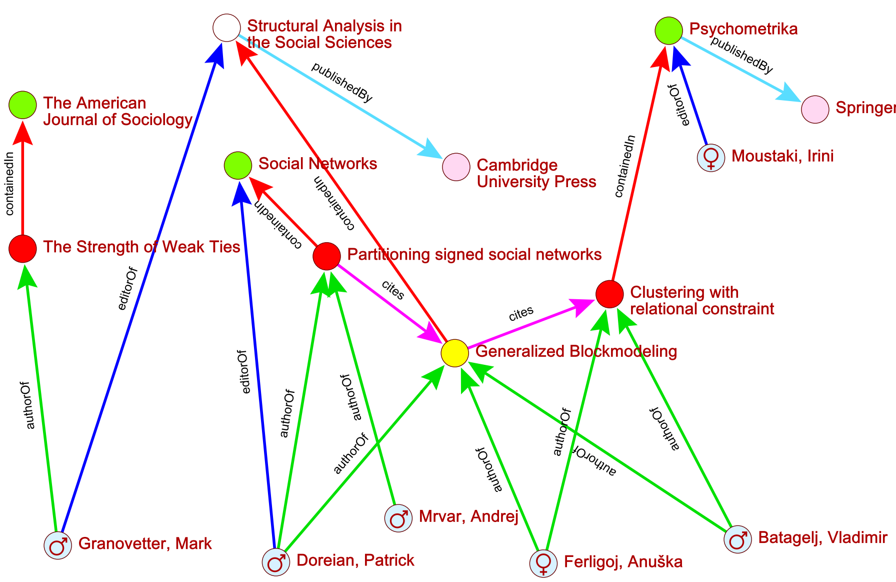

Fig. 5. Bibliographic network – picture / Pajek

Using a short program in R (see Figure 3) we transform both tables into Pajek files: a network file `bib.net` (see Figure 4) and partition files `bibMode.clu` and `bibSex.clu`. All the files related to the bibliographic example are available at GitHub/Bavla [11].

3 Nets and NetsJSON

We were satisfied with the "traditional" network description until we became interested in networks with node/link properties that are not measured in standard scales (ratio, interval, ordinal, nominal), but have structured values (text, subset, interval, distribution, time series, temporal quantity, function, etc.). For describing temporal networks we initially, extending the Pajek format, defined and used the Ianus format [7]. We needed a format that could describe structured values. We could base our format on two options – XML and JSON. We chose JSON and in 2015 started developing the NetJSON format and the Nets Python package to handle networks with structured-valued properties [1,12,13].

On February 26, 2019, the format was renamed to NetsJSON because of the collision with http://netjson.org/rfc.html. NetsJSON has two versions: a *basic* and a *general* version. The current implementation of the Nets library supports only the basic version.

In addition to describing networks with structured values, NetsJSON is expected to offer the capabilities of (most) existing network description formats [8] (archiving, conversion) and provide input data for D3.js visualizations.

3.1 Informal description of the basic NetsJSON format

```
{
"netsJSON": "basic",
"info": {
   "org":1, "nNodes":n, "nArcs":mA, "nEdges":mE,
   "simple":TF, "directed":TF, "multirel":TF, "mode":m,
   "network":fName, "title":title,
   "time": { "Tmin":tm, "Tmax":tM, "Tlabs": {labs} },
   "meta": [events], ...
   },
"nodes": [
   { "id":nodeId, "lab":label, "x":x, "y":y, ... },
   ***
   ]
"links": [
   { "type":arc/edge, "n1":nodeID1, "n2":nodeID2, "rel":r, ... }
   ***
   ]
}
```

where ... are user-defined properties and *** is a sequence of such elements.

An event description can contain the following fields:

```
{  "date": date,
   "title": short description,
   "author": name,
```

```
    "desc": long description,
    "url": URL,
    "cite": reference,
    "copy": copyright
}
```

It is intended to provide information about the "life" of the dataset – changes, releases, uses, publications, etc.

For describing temporal networks a node element and a link element have an additional required property `tq` – a temporal quantity.

For an example see GitHub/Bavla/Graph/JSON/violenceU.json describing Franzosi's violence network.

The general NetsJSON format is also expected to support the description of network collections.

In recent years we also analyzed bike systems (link weight is a daily number of trips distribution), bibliographies (yearly distributions of publications or citations), and multiway networks [3–5]. It turned out that it was necessary to add another main field, `data`, to the basic NetsJSON format, in which we provide additional data about the properties of values (translations of labels in selected languages, algebraic structure [2]).

4 Elements of a common network format

Our experience with network analysis to date is summarized in the following recommendations on the elements of a common format for describing networks.

For data integrity, it makes sense to combine data and metadata into a single file. To preserve the structure of data, it makes sense to base the format on JSON, which fits well with the data structures of modern programming languages.

We would also encourage the provision, as metadata, of information about the context of the network, additional knowledge about it, articles or notebooks on its analysis, comments of users, etc. Kaggle is a good example. An improved ICON repository or Networkrepository (we disagree with their "citation request") could be the way to go. Existing metadata standards should be taken into account (Dublin Core, FAIR, Schema). Data has a "life". When selecting data, its age is often important. Metadata should include at least the creation date and the last modification date.

The format must support all types of networks (simple, 2-mode, linked, multi-relational, multi-level, temporal). The network can contain both arcs and edges, as well as parallel links.

As mentioned earlier, using factorization produces a more concise description of the network. In cases where the node names are not too long and are readable, we sometimes want to avoid factorization. This can be achieved by using a switch that indicates whether factorization is used. We can also shorten the description length by introducing default values. If we also allow counting from 0, it makes sense to add information about the smallest index.

Long labels cause problems when printing/visualizing (parts of) networks and results. Therefore, it is useful to have abbreviated versions of labels available.

Most of the network datasets produced by network science have no node labels. Node labels are not needed if you study distributions, but they are very important in the interpretation of the obtained “important substructures”. We would encourage providing node labels, or at least some typology info in the case of privacy issues.

Operations and transformations on networks: extensions (new yearly data), constructive description (, NetML [6]), intersection, union, ... product; derived networks

We have not yet started working on a general format. It is supposed to enable descriptions of collections of networks. The question arises about the scope of validity of IDs - does the same ID in different networks represent the same or different units? This is important for operations such as union or intersection of networks. Which way to go - introducing contexts or using matchings?

Generalizations: multiway, hypernets.

Additional ideas may be found on the page ”*A Python Graph API?*” [16].

5 Conclusions

The availability of the data used in the article enables the reproducibility and verifiability of the analyses performed. In addition, the obtained results can be verified or supplemented with other methods. When developing new methods, accessible and well-documented data are also very important - it is good to test a new method on several data sets and check whether it gives meaningful/expected results.

5.1 Acknowledgments

The computational work reported in this paper was performed using programs R and Pajek [9]. The code and data are available at Github/Bavla [11].

This work is supported in part by the Slovenian Research Agency (research program P1-0294, research program CogniCom (0013103) at the University of Primorska, and research projects J5-2557, J1-2481, and J5-4596), and prepared within the framework of the COST action CA21163 (HiTEc).

References

1. Batagelj, V.: Network visualization based on JSON and D3.js. `https://github.com/bavla/netsJSON/blob/master/doc/netVis.pdf` (2016), Second European Conference on Social Networks. June 14-17, 2016, Paris. Slides
2. Batagelj, V.: Semirings in network data analysis / an overview. `https://github.com/bavla/semirings/blob/master/docs/semirings.pdf` (2021), Mathematics for Social Sciences and Arts / Algebraic Modeling. MS^2A^2M 2021, on Zoom, May 24-26, 2021. Slides
3. Batagelj, V.: Cores in multiway networks. Social Network Analysis and Mining **14**(1), 122 (2024)

4. Batagelj, V., Ferligoj, A.: Symbolic network analysis of bike sharing data / Citi Bike. https://github.com/bavla/Bikes/blob/master/bikes.pdf (2016), CM-Statistics 2016, Sevilla, 9-11. December 2016. Slides
5. Batagelj, V., Maltseva, D.: Temporal bibliographic networks. Journal of Informetrics **14**(1), 101006 (2020)
6. Batagelj, V., Mrvar, A.: INSNA Sunbelt International Social Network Conference, London, July 6-10 1995. Slides. https://github.com/bavla/netsJSON/blob/master/doc/snetml.pdf (1995), accessed: 2025-03-18
7. Batagelj, V., Praprotnik, S.: An algebraic approach to temporal network analysis based on temporal quantities. Social Network Analysis and Mining **6**, 1–22 (2016)
8. Bodlaj, J., Cerinšek, M.: Network Data File Formats, pp. 1525–1540. Springer New York, New York, NY (2018). https://doi.org/10.1007/978-1-4939-7131-2_298, https://doi.org/10.1007/978-1-4939-7131-2_298
9. De Nooy, W., Mrvar, A., Batagelj, V.: Exploratory social network analysis with Pajek, vol. 46. Cambridge university press (2018)
10. Gephi: Supported graph formats. https://gephi.org/users/supported-graph-formats/ (2022), accessed: 2025-03-18
11. GitHub/Bavla: Bibliographic network example. https://github.com/bavla/netsJSON/tree/master/example/bib (2016), accessed: 2025-03-18
12. GitHub/Bavla: Nets – Python package for network analysis. https://github.com/bavla/Nets (2016), accessed: 2025-03-18
13. GitHub/Bavla: NetsJSON – a JSON format for network analysis. https://github.com/bavla/netsJSON (2016), accessed: 2025-03-18
14. Neal, Z.P., et al.: Recommendations for sharing network data and materials – endorsement page. https://www.zacharyneal.com/datasharing (2024), accessed: 2025-03-18
15. Neal, Z.P., Almquist, Z.W., Bagrow, J., Clauset, A., Diesner, J., Lazega, E., Lovato, J., Moody, J., Peixoto, T.P., Steinert-Threlkeld, Z., et al.: Recommendations for sharing network data and materials. Network Science **12**(4), 404–417 (2024). https://doi.org/10.1017/nws.2024.16
16. The Python Wiki: A Python Graph API? https://wiki.python.org/moin/PythonGraphApi (2011), accessed: 2025-03-18