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 18, 2025 at 04 : 56

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

Keywords: First keyword · Second keyword · Third keyword

1 Introduction

1.1 Software support for network analysis

There are many tools and programs for network analysis UCINET, Pajek, Gephi, Cytoscape, NodeXL, Tulip, GraphViz,

Programmers can use network analysis packages/libraries in programming languages

- **Python:** NetworkX, igraph, Snap.py, graph-tool, NetworKit, PyGraphistry, Nets, cdlib, node2vec, DGL, PyG, Tulip, PyVis,
- **R:** igraph, statnet, sna, qgraph, 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.

Gephi – Supported Graph Formats
GEXF 1.3 primer
Cons matrix – large networks
Cons neighbors – node properties
Tomaž Pisanski et al. – Vega
Primož Potočnik et al. – catalogues

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.

```
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

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.

```
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 1 "',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

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.

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 https://github.com/bavla/netsJSON/tree/master/example/bib.

```
*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

3 Nets and NetsJSON

For dealing with networks with properties with structured values (for example, temporal quantities) we are developing a Python package Nets [**?**].

For describing temporal networks we initially, extending Pajek format, defined and used the Ianus format.

In 2015 we started to develop a new format based on JSON – we named it netJSON. 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. Current implementation of the Nets / TQ library supports only the basic version. Nets.

Besides for a *description* of networks with structured values, NetsJSON should *envelope* (most of) existing network description formats [**?**] (archiving, conversion) and provide input data for D3.js *visualization*s.

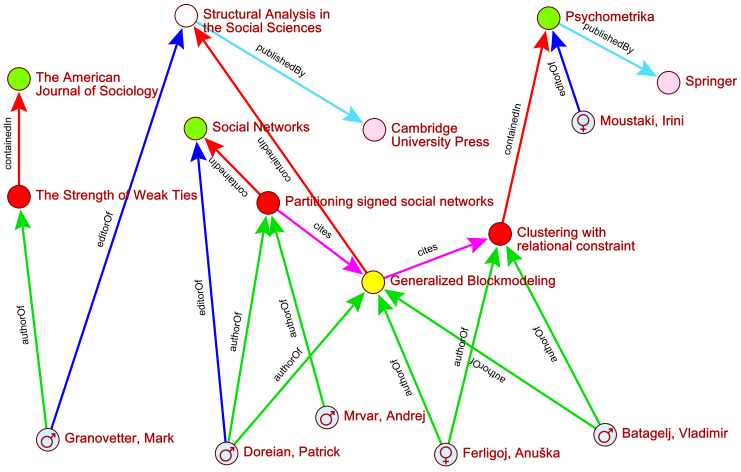

Fig. 5. Bibliographic network – picture / Pajek

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.

Basic netsJSON format An event description can contain fields:

```
{  "date": date,
   "title": short description,
   "author": name,
   "desc": long description,
   "url": URL,
   "cite": reference,
   "copy": copyright
}
```

for describing temporal networks a node element and a link element has an additional required property `tq`

Example 1, Franzosi's violence network / UTF-8 no sig

4 Network formats

4.1 Format

Mixed links directed/undirected

Multiple links

Additional data about values, algebraic structures

Metadata (Dublin Core, FAIR, Schema)

Python page on network DS Python Patterns - Implementing Graphs

!!! Time publication date creation/last change

Values – semiring weights (Semirings)

Structured values: TQs, records, distributions

Values – functions: PERT, circuits, modeling

IDs - in network science often missing; individuals/classes -is a-

Alternative labels for display (short, other languages)

Collections of networks

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

Contexts !!! – matching IDs in different contexts

Generalizations: multiway, hypernets, bikes

Size: huge networks

Metadata, updates, comments

Default values

KG ¡-¿ networks (single relation, type of units)

5 Conclusions

1.
2.
3.

5.1 Acknowledgments

The computational work reported in this paper was performed using a collection of R functions `bibmat` and the program Pajek for analysis of large networks [**?**]. The code and data are available at Github/Bavla [**?**].

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. Arroyuelo, D., Hogan, A., Navarro, G., Reutter, J., Vrgoč, D.: Tackling challenges in implementing large-scale graph databases. Communications of the ACM **67**(8), 40–44 (2024)
2. Batagelj, V.: Social network analysis, large-scale (2009)
3. Batagelj, V.: Analysis and visualization of large networks (20/21 October 2005), slides
4. Chen, L., Zhang, H., Chen, Y., Guo, W.: Blank nodes in RDF. J. Softw. **7**(9), 1993–1999 (2012)
5. Ehrlinger, L., Wöß, W.: Towards a definition of knowledge graphs. SEMANTiCS (Posters, Demos, SuCCESS) **48**(1-4), 2 (2016)
6. Hartig, O., Champin, P.A., Kellogg, G., Seaborne, A. (eds.): RDF 1.2 Concepts and Abstract Syntax (25 February 2025), `https://www.w3.org/TR/rdf12-concepts/`, W3C Working Draft
7. Hayes, J., et al.: A graph model for RDF. Darmstadt University of Technology/University of Chile (2004), `https://users.dcc.uchile.cl/~cgutierr/papers/rdfgraphmodel.pdf`
8. Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., de Melo, G., Gutiérrez, C., Kirrane, S., Labra Gayo, J.E., Navigli, R., Neumaier, S., Ngonga Ngomo, A.C., Polleres, A., Rashid, S.M., Rula, A., Schmelzeisen, L., Sequeda, J.F., Staab, S., Zimmermann, A.: Knowledge Graphs. No. 22 in Synthesis Lectures on Data, Semantics, and Knowledge, Springer (2021). `https://doi.org/10.2200/S01125ED1V01Y202109DSK022`, `https://kgbook.org/`
9. Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., Melo, G.D., Gutierrez, C., Kirrane, S., Gayo, J.E.L., Navigli, R., Neumaier, S., et al.: Knowledge graphs. ACM Computing Surveys (Csur) **54**(4), 1–37 (2021)
10. Hogan, A., Dong, X.L., Vrandečić, D., Weikum, G.: Large language models, knowledge graphs and search engines: A crossroads for answering users' questions. arXiv preprint arXiv:2501.06699 (2025)
11. Nguyen, V., Leeka, J., Bodenreider, O., Sheth, A.: A formal graph model for RDF and its implementation. arXiv preprint arXiv:1606.00480 (2016), `https://arxiv.org/abs/1606.00480`
12. Paulheim, H.: Knowledge graph refinement: A survey of approaches and evaluation methods. Semantic web **8**(3), 489–508 (2016)
13. Tomaszuk, D., Hyland-Wood, D.: RDF 1.1: Knowledge representation and data integration language for the web. Symmetry **12**(1), 84 (2020)
14. Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.W., da Silva Santos, L.B., Bourne, P.E., et al.: The fair guiding principles for scientific data management and stewardship. Scientific data **3**(1), 1–9 (2016)
15. Zhong, L., Wu, J., Li, Q., Peng, H., Wu, X.: A comprehensive survey on automatic knowledge graph construction. ACM Computing Surveys **56**(4), 1–62 (2023)
16. Zloch, M., Acosta, M., Hienert, D., Conrad, S., Dietze, S.: Charaterizing RDF graphs through graph-based measures–framework and assessment. Semantic Web **12**(5), 789–812 (2021)