The Journal of Open Source Software

textNet: Directed, Multiplex, Multimodal Event Network Extraction from Textual Data

Elise Zufall[1] **and Tyler Scott**[1]

1 UC Davis Department of Environmental Science and Policy

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Introduction

Network measurement in social science typically relies on data collected through surveys and interviews. Document-based measurement is automatable and scalable, providing opportunities for large scale or longitudinal research that are not possible through traditional methods. A number of tools exist to generate networks based on co-occurrence of words within documents (such as the Nocodefunctions app (Levallois et al., 2012), the "textnets" package (Bail, 2024), InfraNodus (Paranyushkin, 2018), and many more). But there is, to our knowledge, no open-source tool that generates network data based on the syntactic relationships between entities within a sentence. *textNet* allows a user to input one or more PDF documents and create arbitrarily complex directed, multiplex, and multimodal network graphs. *textNet* also works on arbitrarily long documents, making it well suited for research applications using long texts such as government planning documents, court proceedings, regulatory impact analyses, and environmental impact assessments.

Statement of Need

Network extraction from documents has typically required manual coding. Furthermore, existing network extraction methods that use co-occurrence leave a vast amount of data on the table, namely, the rich edge attribute data and directionality of each verb phrase defining the particular relationship between two entities, and the respective roles of the entity nodes involved in that verb phrase. We present an R package, *textNet*, designed to enable directed, multiplex, multimodal network extraction from text documents through syntactic dependency parsing, in a replicable, automated fashion for collections of arbitrarily long documents. The *textNet* package facilitates the automated analysis and comparison of many documents, based on their respective network characteristics. Its flexibility allows for any desired entity categories, such as organizations, geopolitical entities, dates, or custom-defined categories, to be preserved.

Directed Graph Production

As a syntax-based network extractor, *textNet* identifies source and target nodes. This produces directed graphs that contain information about network flow. Methods based on identifying co-occurring nodes in a document, by contrast, produce undirected graphs. textNet also allows the user to code ties based on co-occurrence in a designated piece of text if desired.

Multiplex Graph Output

Syntax-based measurement encodes edges based on subject-verb-object relationships. *textNet* stores verb information as edge attributes, which allows the user to preserve arbitrarily complex topological layers (of different types of relationships) or customize groupings of edge types to simplify representation.

Multimodal Graph Output

Multimodal networks, or networks where there are multiple categories of nodes, have common use cases such as social-ecological network analysis of configurations of actors and environmental features. Existing packages such as the manynet package (Hollway, 2024) provide analytical functions for multimodal network statistics. *textNet* provides a structure for tagging and organizing arbitrarily complex node labeling schemes that can then be fed into packages for multi-node network statistical analysis. Node labels can be automated (e.g., the default entity type tags for an NLP engine such as *spaCy* (Honnibal et al., 2021)), customized using a dictionary, or based on a hybrid scheme of default and custom labels. Any node type is possible (e.g., species, places, people, concepts, etc.) so this can be adapted to domain specific research applications by applying dictionaries or using a custom NER model.

Avoids Saturation

Co-occurrence graphs have the tendency to generate saturated subgraphs, since every co-occurring collection of entities has every possible edge drawn amongst them. By contrast, *textNet* draws connections not between every entity in the document or even the sentence, but specifically between pairs of entities that are mediated by an event relationship. This leads to sparser graphs that preserve the ability for greater structural variance, and correspondingly, network analysis of structural attributes of the graphs.

Installation

The stable version of this package can be installed from Github, using the *devtools* package (Wickham et al., 2022):

```
devtools::install_github("ucd-cepb/textnet")
```

The *textNet* package suggests several convenience wrappers of packages such as *spacyr* (Benoit et al., 2023), *pdftools* (Ooms, 2024), *igraph* (Csárdi et al., 2024), and *network* (Butts et al., 2023). To use the full functionality of *textNet*, such as pre-processing tools and post-processing analysis tools, we recommend installing these packages, which for *spacyr* requires integration with Python. However, the user may wish to preprocess and parse data using their own NLP engine, and skip directly to the textnet_extract() function, which does not depend on any of the aforementioned packages. The textnet_extract() function does, however, use functions from *pbapply* (Solymos et al., 2023), *data_table* (Barrett et al., 2024), *dplyr* (Wickham et al., 2023), and *tidyr* (Wickham et al., 2024).

Overview and Main Functions

The package architecture relies on four sets of functions around core tasks:

- [OPTIONAL] Pre-processing: pdf_clean(), a wrapper for the pdftools::pdf_text() function which includes a custom header/footer text removal feature; and parse_text(), which is a wrapper for the *spacyr* package and uses the *spaCy* natural language processing engine (Honnibal et al., 2021) to parse text and perform part of speech tagging, dependency parsing, and named entity recognition (NER). Alternatively, the user can skip this step and load parsed text directly into the package. *textNet* is designed for modularity with respect to pdf-to-text conversion and NLP engine. The user can derive plain text by any approach, and likewise perform event extraction with any NLP engine or large language model (LLM) (more on LLM extensions below) and bring these data to *textNet*. The textnet_extract() function expects the parsed table to follow specific conventions for column names and speech tagging, so externally produced data must be converted to standards outlined in the package manual.
- Network extraction: textnet_extract(), which generates a graph database from parsed text based upon tags and dependency relations. The object returned from textnet_extract()

consists of a nodelist, an edgelist with a rich set of edge attributes, a verblist, and a list of potential synonyms for disambiguation.

- Disambiguation: tools for cleaning, recoding, and aggregating node and edge attributes, such as the find_acronyms() function, which can be paired with the disambiguation() function to identify acronyms in the text and replace them with the full entity name.
- Exploration: the export_to_network() function for exporting the graph database to igraph and network objects, top_features() for viewing node and edge attributes, and combine_networks() for aggregating multiple document-based graphs based on common nodes.

The figure below summarizes the functionality of *textNet* and the flow of function outputs. Optional data cleaning features are shown with dotted arrows.

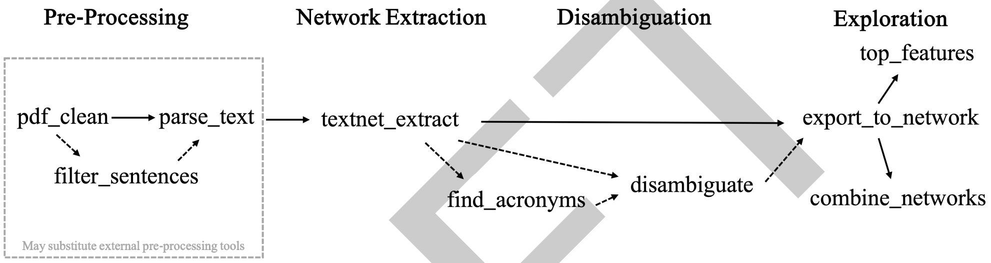

Figure 1: Workflow of textNet Functions

Applications

textNet has applications in governance network scholarship, as demonstrated by Zufall and Scott (2024) and by ongoing work on groundwater governance at the UC Davis Center for Environmental Policy and Behavior. Additional potential applications include legal scholarship, social-ecological network analysis, government planning documents, court proceedings, archival research, communication and media research, and other fields interested in exploring events and entity relationships in textual data.

Potential Further Analyses

textNet is compatible with standard network analysis tools in R. Functionality from *ggraph* (Pedersen & RStudio, 2024), *sna* (Butts 2024), *igraph* (Csárdi et al. 2024), *network* (Butts et al. 2023), and other network visualization and analysis packages can be used to further explore the extracted networks.

The *ggraph* package has been used to create the network visualization seen here, using a weighted version of a igraph constructed using the "old_new_parsed" sample data in *textNet*.

The Journal of Open Source Software

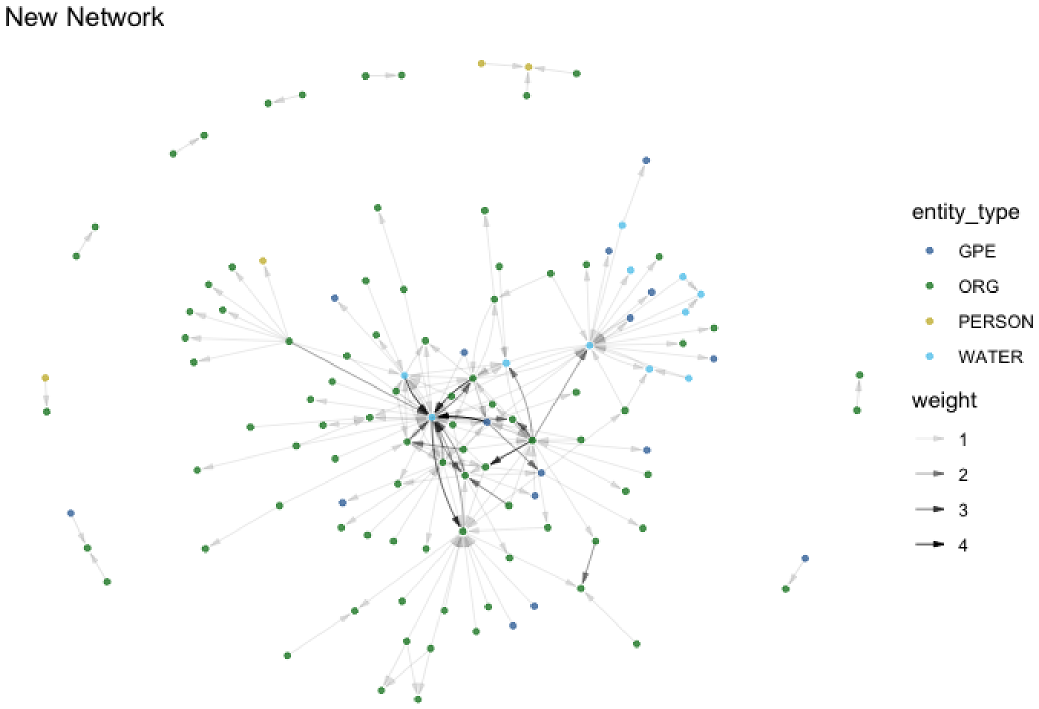

Figure 2: Representation of the Event Network of the New Plan

The network-level attributes output from export_to_network can also be analyzed against exogenous metadata that has been collected separately by the researcher regarding the different documents and their real-world context. The extracted networks, with their collections of verb attributes, node attributes, edge incidences, and edge attributes, can also be analyzed through a variety of tools, such as an Exponential Random Graph Model, to determine the probability of edge formation under certain conditions. A Temporal Exponential Random Graph Model could also shed light on the changes of a document over time, such as the multiple versions of the groundwater sustainability plan in this example.

Vignette

More information about the entity network extraction algorithm and an example start-to-finish data processing and analysis workflow can be found in the vignette for this package. The vignette uses sample data that travels with the *textNet* package.

Acknowledgements

The authors gratefully acknowledge the support of the Sustainable Agricultural Systems program, project award no. 2021-68012-35914, from the U.S. Department of Agriculture's National Institute of Food and Agriculture and the National Science Foundation's Dynamics of Integrated Socio-Environmental Systems program, grant no. 2205239.

Appendix

This appendix describes the pre-processing tools available through the *textNet* package, which enable the user to generate the data frame expected by the textnet_extract() function.

References

Bail, C. (2024). *Cbail/textnets* (Version 0.1.1). https://github.com/cbail/textnets

Barrett, T., Dowle, M., Srinivasan, A., Gorecki, J., Chirico, M., Hocking, T., Schwendinger, B., Stetsenko, P., Short, T., Lianoglou, S., Antonyan, E., Bonsch, M., Parsonage, H., Ritchie, S., Ren, K., Tan, X., Saporta, R., Seiskari, O., Dong, X., … Krylov, I. (2024). *Data.table: Extension of 'data.frame'* (Version 1.16.2). https://cran.r-project.org/web/packages/data.table/index.html

Benoit, K., Matsuo, A., Gruber, J., & Council (ERC-2011-StG 283794-QUANTESS), E. R. (2023). *Spacyr: Wrapper to the 'spaCy' 'NLP' library* (Version 1.3.0). https://cran.r-project.org/web/packages/spacyr/index.html

Butts, C. T., Hunter, D., Handcock, M., Bender-deMoll, S., Horner, J., Wang, L., Krivitsky, P. N., Knapp, B., Bojanowski, M., & Klumb, C. (2023). *Network: Classes for relational data* (Version 1.18.2). https://cran.r-project.org/web/packages/network/index.html

Csárdi, G., Nepusz, T., Traag, V., Horvát, S., Zanini, F., Noom, D., Müller, K., Salmon, M., Antonov, M., & details, C. Z. I. igraph author. (2024). *Igraph: Network analysis and visualization* (Version 2.1.1). https://cran.r-project.org/web/packages/igraph/index.html

Hollway, J. (2024). *Manynet: Many ways to make, modify, map, mark, and measure myriad networks* (Version 1.2.6). https://CRAN.R-project.org/package=manynet

Honnibal, M., Montani, I., Van Landeghem, S., & Boyd, A. (2021). *spaCy: Industrial-strength natural language processing in python* (Version 3.1.3). https://github.com/explosion/spaCy/tree/master

Levallois, C., Clithero, J. A., Wouters, P., Smidts, A., & Huettel, S. A. (2012). Translating upwards: Linking the neural and social sciences via neuroeconomics. *Nature Reviews Neuroscience*, *13*(11), 789–797. https://nocodefunctions.com/cowo/semantic_networks_tool.html

Ooms, J. (2024). *Pdftools: Text extraction, rendering and converting of PDF documents* (Version 3.4.1). https://cran.r-project.org/web/packages/pdftools/index.html

Paranyushkin, D. (2018). *InfraNodus*. Nodus Labs. https://infranodus.com/

Pedersen, T. L., & RStudio. (2024). *Ggraph: An implementation of grammar of graphics for graphs and networks* (Version 2.2.1). https://cran.r-project.org/web/packages/ggraph/index.html

Solymos, P., Zawadzki, Z., Bengtsson, H., & Team, R. C. (2023). *Pbapply: Adding progress bar to '*apply' functions* (Version 1.7-2). https://cran.rstudio.com/web/packages/pbapply/index.html

Wickham, H., François, R., Henry, L., Müller, K., Vaughan, D., Software, P., & PBC. (2023). *Dplyr: A grammar of data manipulation* (Version 1.1.4). https://cran.r-project.org/web/packages/dplyr/index.html

Wickham, H., Hester, J., Chang, W., Bryan, J., & RStudio. (2022). *Devtools: Tools to make developing r packages easier* (Version 2.4.5). https://cran.r-project.org/web/packages/devtools/index.html

Wickham, H., Vaughan, D., Girlich, M., Ushey, K., Software, P., & PBC. (2024). *Tidyr: Tidy messy data* (Version 1.3.1). https://cran.r-project.org/web/packages/tidyr/index.html

Zufall, E., & Scott, T. A. (2024). Syntactic measurement of governance networks from textual data, with application to water management plans. *Policy Studies Journal*, *n/a*. https://doi.org/10.1111/psj.12556