Reddit Network Toolkit (rnt):
A simple python package for generating and analyzing Reddit networks

Jacob Rohde, PhD, MPH[1]

[1]Cancer Prevention Fellowship Program, Health Communication and Informatics Research Branch, Division of Cancer Control and Population Sciences, National Cancer Institute

Corresponding author at:
Correspondence concerning this article should be addressed to Jacob A. Rohde, Cancer Prevention Fellowship Program, Health Communication and Informatics Research Branch, Division of Cancer Control and Population Sciences, National Cancer Institute, 9609 Medical Center Drive, Bethesda, MD 20850-9761, USA. E-mail: jacob.rohde@nih.gov | Phone: (240) 275-6638

Role of funding sources:
The author reports no funding source that supported this research.

Declaration of conflicts:
None.

Abstract

Public data from the poplar social media platform Reddit offer communication researchers an opportunity to investigate a host of important and timely scientific questions through a social network analysis framework. The current paper discusses a simple-to-use, open-source Python package called '*rnt*'—or the Reddit Network Toolkit—designed to extract social network data from general or subreddit-specific keyword queries. The current paper provides an overview of the key features of the package, including how to generate various forms of Reddit network data sets (e.g., directed/undirected adjacency matrices, edge lists), compute simple descriptive statistics, and visualize network sociograms. I also provide several code snippet examples of how to use *rnt* through multiple case studies analyzing vaping discussion networks on Reddit, and offer suggestions about using *rnt* with other data analysis Python packages. I conclude by discussing various *rnt* applications in research and practice, and I address some ethical considerations that should be raised at the onset of any project investigating social media data.

Keywords: Reddit, social media, social network analysis, big data, Python, open source.

Introduction

The availability of public social media data has helped develop the communication research landscape. Studies have used social media data to extend prominent theories and frameworks, such as agenda-setting and two-/multi-step flow, and to explore public reactions to emerging crises (e.g., international conflict, global health pandemics) (Boon-Itt & Skunkan, 2020; Comunello & Anzera, 2012; Gilardi et al., 2021; Guo et al., 2018; Park, 2013; Vargo et al., 2018). Given the utility of investigating social media discourse, it is important that researchers have the tools to be able to analyze data from these platforms.

There are several open-source software packages that can extract social media data. Many of these packages rely on access to an application programming interface (API). Briefly, an API is a web-based service—typically provided by private companies or organizations—that allows individuals to interact with a given software and its data (Lomborg & Bechmann, 2014). A popular social media example is Twitter's API which, at the time of writing this, allows academic researchers who sign up for a Twitter developer account to stream real time data and download up to 10 million historic tweets per month for free through their academic v2 API endpoint (Twitter, n.d.). Example open-source packages for downloading API Twitter data include *Tweepy* for Python and *rtweet* for R (Kearney, 2019; Roesslein, 2022). There are also thousands of additional third-party open-source packages across different programming languages for analyzing tweets. For a detailed computational overview of working with Twitter data, see Jürgens and Jungherr (2016).

Reddit is another platform that allows public access to their API. Briefly, Reddit is a thread-based social media website, and a distinguishing feature of this platform is its use of subreddits, which are user-created, topic-based communities (Anderson, 2015). Reddit encourages, if not predicates, participation in subreddits. This is a useful feature for communication research as it provides a built-in index for examining online dialogue about a specific topic, such as subreddits for cancer support or political partisanship. In addition, the thread-based participation structure of the platform makes it useful for examining interpersonal communication among users and for modeling online discussion networks. For example, research could use data from Reddit to identify and track how misinformation both originates and spreads across different communities on the platform.

In the current paper, I introduce the Reddit Network Toolkit (*rnt*). This Python package was developed as a tool for social scientists to easily extract and analyze online discussion networks from Reddit. I created this package for two reasons. First, *rnt* emphasizes social network analysis, and the package is capable of generating edge lists, adjacency matrices and other network-focused data sets that are derived from Reddit users' participation in online discussion threads or subreddit communities. This approach is different than other popular Python packages that exist for collecting Reddit data such as *praw* (Boe, 2016), as these packages typically only generate data that match keyword search terms (e.g., extracting all Reddit submissions or comments that contain the word "cancer").

Second, there is a dearth of published communication research investigating Reddit. A recent systematic review found only 132 published articles across all humanity and social science disciplines reporting data from Reddit since 2010 (Proferes et al., 2021). Of those, only 13 articles were from communication or journalism subfields. Moreover, a title and abstract search for the term "Reddit" in the Communication & Mass Media Complete database at the time of writing this paper in December 2022 returned 157 results. By contrast, searching "Twitter" or "Facebook" in the same database returned 3,339 results. Recent national data from the Pew Research Center found that nearly one-fifth of U.S. adults reported having ever used Reddit, which is an increase from 11% in 2019 (Pew Research Center, 2021). This growth speaks to the platform's popularity and the need for researchers to consider Reddit as an important hub for online public engagement and discourse. *rnt* aims to be a springboard to increase research about this platform.

rnt was designed to be accessible for those with a minimal background in computer programming. Its primary goal is to make it easy for researchers to quickly generate a multitude of different data sets and to investigate research questions with only a few lines of code. That said, I do assume readers (and users of the package) have some understanding of the Python programming language prior to using *rnt*, such as knowing how to install and import third-party packages, as well as basic language syntax. I recommend *Automate the Boring Stuff With Python* by Sweigart (2019) for a useful beginner's guide to learning Python.

The remaining sections of this paper provide a brief overview of some key terminology associated with Reddit, followed by a top-level introduction to the features of *rnt*. I provide examples of how to use *rnt* via several code snippets. These examples are set in the context of vaping discussion networks and were informed by previous work using *rnt* (Rohde et al., under review); however, this package can be used to investigate a host of other topics. Notably, both code snippets and in-text references to python code, such as listing out *rnt* functions and arguments, are formatted using Consolas font for clarity. I conclude with a discussion about various uses and limitations of *rnt*, and I highlight some key ethical considerations that should be raised at the onset of any research project using this (or a similar) package.

Reddit terminology

Reddit has several features that set it apart from other social media platforms. This section provides an overview of some helpful terminology to know when working with Reddit data. First, people who use Reddit are referred to as either Redditors or Reddit users. As mentioned above, Redditors predominately interact with one another in subreddits, which are user-created, topic-based communities. For example, the subreddit r/stopsmoking is a public community where Redditors can provide one another with smoking cessation-related support and advice. Content posted to subreddits are typically monitored by one or more moderators to ensure Redditors follow community rules (e.g., only talking about the topic of the subreddit, no profanity, etc.). Most all Reddit participation occurs in subreddits; though, Redditors can also choose to post content to their user profile page. Reddit uses a thread-based model to organize its participation. Posts that initiate a thread in a subreddit or on a Redittor's profile is called a submission, and the content posted in response to submissions are called comments. Reddit submissions can contain media, such as images or hyperlinks, but comments are restricted to text. Finally, both Reddit submissions and comments can be evaluated by peers via an up-/down-vote feature. Content with high up- versus down-vote ratios are typically featured toward the top of subreddits or submission threads.

Installing *rnt*

rnt is a free-to-use, open-source Python package that can be downloaded via the preferred installer program (pip) and the Python Packaging Index (see https://pypi.org/project/rnt/) or through the package's GitHub Repository (see https://github.com/jarohde/rnt). The package requires that the *Numpy*, *NetworkX*, *pandas*, and *pmaw* packages are also installed, as well as their respective dependencies. The network visualization function in the package also requires installing *Matplotlib*, but using this function is optional. I Recommend installing *rnt* through pip to ensure Python is using the most stable package version.

Extracting Reddit data

One of *rnt*'s primary uses is to extract Reddit data based on keyword or subreddit search terms. The simplest way to do this is to use `GetRedditData()`. This collection feature will first pull all Reddit submissions that meet search term criteria, and will then pull the corresponding comments of each of the submissions (where applicable). The result of this extraction process is a data set with inherent network properties linking Redditors who are engaging in online discussions with one another. `GetRedditData()` accepts the following arguments:

- **`search_term:`** The only required argument; takes a string (or list of strings) as a keyword for searching Reddit submissions. If choosing to extract data from a specific subreddit instead of a general keyword query, only list one subreddit name verbatim as a string in this parameter (e.g., `search_term='AskReddit'`).

- **`search_term_is_subreddit:`** Optional Boolean (True or False) argument to signify whether `GetRedditData()` extracts a subreddit data set; default set to False. If set to True, data from only one subreddit can be extracted per object instance.

- **`size:`** Optional integer argument to signify how many Reddit submissions and their associated comments to extract; default set to 500 submission. `GetRedditData()` should only be used to extract limited or exploratory data sets. I recommend using the Pushshift Reddit repository for extracting large data sets.

- **`end_date/start_date:`** Optional string arguments for `GetRedditData()` to signify a data collection period; default end date set to current date and default start date set to one week prior. Format should be strings structured like `'YYYY, MM, DD'` (e.g., `start_date='2022, 5, 27'` for May 27, 2022).

A `GetRedditData()` object can be saved as a variable, typically called "data," in a Python workspace. From here, users have access to the following object methods and attributes:

- **`GetRedditData.df:`** Attribute to access the Reddit data set as a *pandas* DataFrame object.

- **`GetRedditData.write_data():`** Method that writes the *pandas* DataFrame object to file. The method can take `file_type` and `file_name` as optional arguments. `file_type` indicates what file format to use when writing the data set and accepts a string of either `'json'` or `'csv'`; default file format set to json. `file_name` takes a string to indicate what the file name should be saved as; default name set to the search term provided.

The data set generated from the `GetRedditData.df` attribute includes several metadata variables, such as Reddit author names, comment/submission publication date, and the subreddit community that posts were published in. This makes it useful for researchers to extract a data set about a discussion topic and to compute simple descriptive statistics using Python in a matter of minutes. Researchers can also save the data to file and import it to other statistical software applications, such as R or SPSS. For detailed information about what certain Reddit metadata variables represent in the `GetRedditData.df` data set, see Baumgartner et al. (2020). Code snippet 1 provides an example of how researchers could use *rnt* and `GetRedditData()` to extract a sample data set from the r/electronic_cigarette subreddit, which has more than 200,000 Redditors subscribed to the community.

Generating a network object

This paper assumes readers know basic network analysis concepts such as the difference between edge lists and adjacency matrices, directed/undirected networks, and weighted/unweighted networks. For an overview of these concepts, see Wasserman and Faust (1994), Borgatti et al. (2018), or any other comprehensive book on social network analysis.

As previously mentioned, *rnt* generates networks from Reddit based on data from online discussion threads. In these networks, the nodes are the Redditors, and the connections between nodes are based on submissions and comments. For example, redditor_a commenting to a submission by redditor_b creates a redditor_a → redditor_b directed network tie. If redditor_b also commented on a separate submission by redditor_a, that would create bidirectional network tie (redditor_a ←→ redditor_b). See Figure 1 for an overview of how *rnt* initializes its networks.

rnt can generate directed and undirected edge lists, adjacency matrices, and other types of network data sets using the `GetRedditNetwork()` feature. Moreover, these data sets come with various node and edge-level attributes to help researchers characterize their networks, such as the frequency that two nodes interact with one another (i.e., edge weight), what subreddit communities that online discussion threads occur in, or what user-defined keywords might be present or absent in these threads. The `GetRedditNetwork()` object takes the following arguments:

- **`reddit_dataset:`** The only required argument, which accepts a Reddit data set or a `GetRedditData()` object.

- **`edge_type:`** Optional string argument of either `'directed'` or `'undirected'` to signify network edge type; default edge type set to directed.

- **`text_attribute:`** Optional string, list, or dictionary argument to characterize an edge attribute based on one or more text categories. Providing the argument with a string or list will compute a single text attribute. Providing the argument with a dictionary will generate multiple text attributes. The result from this argument will be True or False values for a network edge if the Reddit submission initiating the edge contains the provided keyword(s). These data are stored as separate variables that can be accessed via various `GetRedditNetwork()` attributes (discussed below). It should be noted that this argument uses partial string searches, meaning the text attribute keyword "addict" will mark submissions containing words such as "addiction" or "addictive" as True; however, these keywords do not need to be case sensitive (e.g., "addict" is equal to "Addict"). An example multi text attribute argument formatted using a dictionary data type is below:

```
text_attribute={'apples': ['fuji', 'red delicious', 'granny smith'],
                'oranges': ['valencia', 'mandarin', 'tangerine'],
                'berries': ['blueberry', 'raspberry', 'blackberry']}
```

Once the `GetRedditNetwork()` object has been initialized and stored in memory, users will have access to the following attributes and methods:

- **`GetRedditNetwork.edge_list:`** Attribute that returns a *pandas* DataFrame of the network edge list with columns for the submission author (target), commenter (sender), edge weight, the subreddit the edge occurred in, and any additional text attribute columns.

- **`GetRedditNetwork.node_list:`** Attribute that returns a *pandas* DataFrame of the network node list with columns for each unique node (i.e., Redditor), the node's in-, out-, and total degree network values, and a list of subreddits that the node participated in throughout the data set (both as a commenter and submission author).

- **`GetRedditNetwork.graph:`** Attribute that returns a *NetworkX* graph object.

- **`GetRedditNetwork.adjacency:`** Attribute that returns a dictionary of network adjacency matrices. Both weighted and unweighted matrices are returned by default. The dictionary will also return weighted adjacency matrices for each optional edge-based text attribute that users defined when creating the `GetRedditNetwork()` object.

- **`GetRedditData.write_data():`** Object method that writes edge and node list data sets to file. The method takes `file_type`, `file_name`, and `adjacency` as optional arguments. `file_type` indicates what file format to use when writing the data sets and accepts a string argument of either `'json'` or `'csv'`; default file format set to json. `file_name` accepts a string to indicate what name to save the files as. `adjacency` accepts a Boolean and indicates whether to write the data sets as adjacency matrices instead of edge and node lists.

Code snippet 2 provides an example of how to use *rnt* to generate different network data sets using `GetRedditNetwork()`. This example uses the same Reddit data set about e-cigarettes that was extracted in the previous code snippet.

Computing subreddit- and thread-level Reddit statistics

rnt has two functions that compute basic descriptive statistics about a Reddit data set. These functions are `subreddit_statistics()` and `reddit_thread_statistics()`. The only user-defined argument required for both functions is `'reddit_dataset'`, which takes a `GetRedditData()` object (see above). Alternatively, users can pass in their own *pandas* DataFrame if formatted correctly (i.e., same column names as what is generated by `GetRedditData()`). A brief description of these two functions is provided below.

`subreddit_statistics()` computes separate social networks bound by discussions occurring at the subreddit-level. That is, *rnt* subsets the full data set at subreddit-level, and then creates `GetRedditNetwork()` objects for each subreddit community. This function returns a *pandas* DataFrame with each row representing a unique subreddit network. Example variables in this data set include number of submissions and comments, subreddit network density, and mean/median graph degree, among others.

`Reddit_thread_statistics()` computes thread-level descriptive statistics of a Reddit data set. Here, I define a thread as a collection of a single Reddit submission and its corresponding comments (see Figure 1). This function returns a *pandas* DataFrame with each row representing a unique Reddit thread. Example thread-level variables in this data set include submission author (i.e., Redditor who initiated the thread), the subreddit the thread occurred in, and the mean/median comment response time, among others.

Single network plot

rnt includes a simple network plotting function called `single_network_plot()`, which accepts either a `GetRedditNetwork()` or *NetworkX* graph object as its only required argument. This function is meant for generating simple sociograms to get a sense of the Reddit network data. `single_network_plot()` is a wrapper function and accepts many of the same optional arguments as the draw feature used by *NetworkX* and *Matplotlib*. Example arguments include `with_labels`, `edge_width`, and `node_color`. In addition, users can specify one of several *NetworkX* plotting algorithms via the `pos` argument, with input options including `"spring_layout"` (i.e., force-directed algorithm; default option) and `"circular_layout"`. For more information about these arguments, see here: https://networkx.org/documentation.

Code snippet 3 provides a brief example of how to use `single_network_plot()`. This example uses the same e-cigarette data set as the previous code snippets. I also provide sample code on how to size network nodes based on their degree centrality scores, as well as how to color edges based on if the submissions discuss one of the two text attributes defined in the previous code snippet. Similar methods can be used to characterize node colors and edge weights in the sociogram.

Additional *rnt* analyses using *pandas*

rnt data can be further explored using other Python data analysis packages. Code snippet 4 presents a brief example of how researchers can use functions and methods from the *pandas* library alongside *rnt*. This example draws from a new data set extracted using multiple vaping-related keywords to search for submissions (and their comments) across the platform rather than focusing on data from a single subreddit. In this example, I subset the data set to submissions and comments published in the top three subreddits (determined by frequency). Next, I compute `reddit_thread_statistics()` and `subreddit_statistics()` on the subset data, and show some example results from these analyses. Note that the `mean_response_time` variable computed in the thread statistics data set is represented in seconds (e.g., 33716 seconds = 9.37 hours). Further, I changed all Redditor usernames in these examples to random 5-character strings to protect their identity. Finally, I used the `single_network_plot()` to print separate sociograms of each of the top subreddits and show what an example edge list data set from these networks looks like. The approach in this code snippet is only a small example of how researchers could investigate data from *rnt*. For more utility, I recommend extracting the graph from a `GetRedditNetwork()` *rnt* object and exploring the various attributes and functions that the *NetworkX* package affords.

Discussion

This paper introduced the open-source Python package *rnt*. The purpose of this package is to provide researchers with a free tool for analyzing data from the social media platform Reddit. Through multiple code snippet examples, this paper demonstrated how *rnt* can be used to extract keyword-specific network data derived from Reddit (or subreddit) discussion threads, generate additional network data sets (e.g., edge lists, adjacency matrices), and compute various statistics on these data. I also showed how to visualize a simple Reddit network sociogram using *rnt*. Given Reddit's substantial growth in popularity over recent years (Pew Research Center, 2021), having readily accessible tools like *rnt* can help communication researchers best investigate and understand this platform's role as a public sphere across a number of important topics.

Perhaps *rnt*'s best feature is its ease of use. The examples in this paper yielded analyzable data sets with only a few lines of simple code. This is important for communication researchers who may be interested in empirically exploring interpersonal discourse on Reddit but do not have a strong background in computer science and Python programming. Furthermore, many of the data sets that *rnt* provides can be exported to other file formats, such as json and csv. Thus, researchers could use *rnt* to first generate data, and then import said data into R, SPSS, or another statistical analysis program. Those familiar with Python, however, can extend the utility of *rnt* by pairing this package with other notable data and network analysis Python packages, such as *pandas* and *NetworkX*.

rnt has several other applications beyond a tool for research. For example, its straightforward code and ability to extract keyword-specific data sets makes *rnt* a useful aid for teaching various communication research methods, such as network analysis, social media analysis, and Python programming. Further, it is important to emphasize that *rnt* is an open-source package, meaning users are free to modify the source code or integrate its many features to support other projects, be it personal or professional.

There are some limitations with *rnt* that should be addressed. First, *rnt*'s `GetRedditData()` feature is built on top of the *pmaw* Python package, which uses the Pushshift API to extract Reddit data. Since these are third-party tools, there is no guarantee that the data collected by *rnt* encompasses the full corpus of Reddit submissions and their associated comments. For example, Pushshift may not be able to index content from private subreddits, meaning these data would not be available through *rnt*. Moreover, *rnt* will not be able to collect data if Pushshift servers experience outages; though, the package's other features will still be useable. Another limitation is that *rnt* was not designed to extract very large data sets. Although one can customize the number of submissions to extract, researchers investigating topics with a presumed large corpus of content (tens of thousands of submissions and comments) should consider directly using the Pushshift database to download and parse monthly data batches (Baumgartner et al., 2020).

Prior to concluding, it is important to discuss some ethical considerations about using and publishing work from *rnt*, and from social media data more generally. Cases such as the now infamous "Tastes, Ties, and Time" Facebook investigation have engendered a dialogue about publishing data from social media platforms (Lewis et al., 2008), and studies have shown mixed public and professional reactions about using such data for research purposes (Golder et al., 2017; Shilton & Sayles, 2016). A central argument in this discussion has to do with the publicly available nature of social media data (Zimmer, 2010). Indeed, *rnt* and many other API open-source packages only extract content from users with public accounts, meaning anyone could feasibly search for and access such information; however, just because the data are public does not mean they are not sensitive. Acknowledging this, Zook et al. (2017) encourage researchers to consider the complexities of social media data, and recommend thinking critically about the potential consequences and discriminatory practices that could arise from singling out data from particular individuals or groups.

One approach to protecting user privacy when working with *rnt* is to anonymize results prior to publication. This practice includes masking user information (e.g., account usernames) and paraphrasing text data (e.g., tweets, Reddit submissions) to prevent the opportunity for reverse identification. Another useful safeguard is to report social media data in aggregate. In the context of Reddit, this could include analyzing broad themes throughout a corpus of submissions and comments rather than publishing specific quotes from individual Redditors. It is also important to consider data sharing practices when working with social media data. While data transparency and accessibility are important for replication science, it may not always be feasible to disseminate data sets to others, especially if they contain sensitive content. As a solution, researchers can acknowledge in their publications that anonymized versions of their data sets may be available to others upon reasonable request. Finally, submitting analytical approaches to local institutional review boards may also allay some ethical concerns.

Going forward, I urge researchers to reflect on any potential consequences that may arise when investigating and publishing social media data. Ethical considerations and best practices outlined by Zook et al. (2017), Conway (2014), Hunter et al. (2018), and others offer useful guidelines and starting points; though, it is important to recognize that each case study is different, and the ethics of researching social media data are dynamic and evolving. This is not to say that data from platforms such as Reddit should never be published, but having these discussions up front go a long way in protecting individual privacy.

Conclusion

Open-source software development is key to advancing social media research. Communication scientists rely on these tools to generate data and test a host of hypotheses. *rnt* contributes to this area of work by providing researchers an easy-to-use tool for analyzing social network data sets from Reddit—an under-researched yet popular and growing social media platform.

References

Anderson, K. E. (2015). Ask me anything: What is Reddit? *Library Hi Tech News*, *32*(5), 1-11. https://doi.org/https://doi.org/10.7282/T3D220BR

Baumgartner, J., Zannettou, S., Keegan, B., Squire, M., & Blackburn, J. (2020). The Pushshift Reddit dataset. International AAAI Conference on Web and Social Media,

Boe, B. (2016). Python Reddit api wrapper (PRAW).

Boon-Itt, S., & Skunkan, Y. (2020). Public perception of the COVID-19 pandemic on Twitter: Sentiment analysis and topic modeling study. *JMIR Public Health Surveillance 6*(4), e21978. https://doi.org/10.2196/21978

Borgatti, S. P., Everett, M. G., & Johnson, J. C. (2018). *Analyzing Social Networks*. Sage.

Comunello, F., & Anzera, G. (2012). Will the revolution be tweeted? A conceptual framework for understanding the social media and the Arab Spring. *Islam and Christian–Muslim Relations*, *23*(4), 453-470.

Conway, M. (2014). Ethical issues in using Twitter for public health surveillance and research: developing a taxonomy of ethical concepts from the research literature. *J Med Internet Res*, *16*(12), e290. https://doi.org/10.2196/jmir.3617

Gilardi, F., Gessler, T., Kubli, M., & Müller, S. (2021). Social Media and Political Agenda Setting. *Political Communication*, *39*(1), 39-60. https://doi.org/10.1080/10584609.2021.1910390

Golder, S., Ahmed, S., Norman, G., & Booth, A. (2017). Attitudes Toward the Ethics of Research Using Social Media: A Systematic Review. *J Med Internet Res*, *19*(6), e195. https://doi.org/10.2196/jmir.7082

Guo, L., A. Rohde, J., & Wu, H. D. (2018). Who is responsible for Twitter's echo chamber problem? Evidence from 2016 U.S. election networks. *Information, Communication & Society*, *23*(2), 234-251. https://doi.org/10.1080/1369118x.2018.1499793

Hunter, R. F., Gough, A., O'Kane, N., McKeown, G., Fitzpatrick, A., Walker, T., McKinley, M., Lee, M., & Kee, F. (2018). Ethical Issues in Social Media Research for Public Health. *Am J Public Health*, *108*(3), 343-348. https://doi.org/10.2105/AJPH.2017.304249

Jürgens, P., & Jungherr, A. (2016). A tutorial for using Twitter data in the social sciences: Data collection, preparation, and analysis. https://doi.org/http://dx.doi.org/10.2139/ssrn.2710146

Kearney, M. W. (2019). rtweet: Collecting and analyzing Twitter data. *Journal of Open Source Software*, *4*(42), 1829.

Lewis, K., Kaufman, J., Gonzalez, M., Wimmer, A., & Christakis, N. (2008). Tastes, ties, and time: A new social network dataset using Facebook.com. *Social Networks*, *30*(4), 330-342. https://doi.org/10.1016/j.socnet.2008.07.002

Lomborg, S., & Bechmann, A. (2014). Using APIs for data collection on social media. *The Information Society*, *30*(4), 256-265.

Park, C. S. (2013). Does Twitter motivate involvement in politics? Tweeting, opinion leadership, and political engagement. *Computers in Human Behavior*, *29*(4), 1641-1648. https://doi.org/10.1016/j.chb.2013.01.044

Pew Research Center. (2021). Social media fact sheet. https://www.pewresearch.org/internet/fact-sheet/social-media/

Proferes, N., Jones, N., Gilbert, S., Fiesler, C., & Zimmer, M. (2021). Studying reddit: A systematic overview of disciplines, approaches, methods, and ethics. *Social Media + Society*, *7*(2), 1-14.

Roesslein, J. (2022). *Tweepy: Twitter API library*. In (Version 4.10.1) https://docs.tweepy.org/en/stable/

Rohde, J. A., Liu, J., & Rees, V. W. (under review). Community and opinion leadership effects on vaping discourse: A network analysis of online Reddit threads. *Journal of Health Communication*.

Shilton, K., & Sayles, S. (2016). *"We Aren't All Going to Be on the Same Page about Ethics": Ethical Practices and Challenges in Research on Digital and Social Media* 2016 49th Hawaii International Conference on System Sciences (HICSS),

Sweigart, A. (2019). *Automate the boring stuff with Python: Practical programming for total beginners*. No Starch Press.

Twitter. (n.d.). *Twitter API: Academic Research access*. https://developer.twitter.com/en/products/twitter-api/academic-research

Vargo, C. J., Guo, L., & Amazeen, M. A. (2018). The agenda-setting power of fake news: A big data analysis of the online media landscape from 2014 to 2016. *New Media & Society*, *20*(5), 2028-2049.

Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications*. Cambridge University Press.

Zimmer, M. (2010). “But the data is already public”: on the ethics of research in Facebook. In K. W. Miller & M. Taddeo (Eds.), *The Ethics of Information Technologies* (pp. 229-241). Routledge.

Zook, M., Barocas, S., Boyd, D., Crawford, K., Keller, E., Gangadharan, S. P., Goodman, A., Hollander, R., Koenig, B. A., Metcalf, J., Narayanan, A., Nelson, A., & Pasquale, F. (2017). Ten simple rules for responsible big data research. *PLoS Comput Biol*, *13*(3), e1005399. https://doi.org/10.1371/journal.pcbi.1005399

Code snippet 1. Extracting Reddit data using *rnt*.

Console input

```
data = rnt.GetRedditData(search_term='electronic_cigarette',
                         search_term_is_subreddit=True,
                         size=50)

print(data)

print('Variables included in the data set:')
print(list(data.df.columns))

data.write_data(file_name='e_cigarette_data',
                file_type='csv')
```

Console output

```
Collecting 50 submissions and their comments from the "electronic_cigarette"
subreddit.

Collecting comments (batch 1 of 1).

Reddit data object:
Search term(s): electronic_cigarette
Search term is subreddit: True
Total dataframe size: 294
Number of Reddit submissions: 50
Number of Reddit comments: 244

Variables included in the data set
['author', 'title', 'selftext', 'body', 'id', 'subreddit', 'parent_id', 'score',
'link_id', 'created_utc', 'created', 'post_type']

Writing e_cigarette_data.csv to file.
```

Code snippet 2. Generate network data sets using *rnt*.

Console input

```
attribute_dictionary = {'addiction': ['addict', 'nicotine', 'dependence'],
                        'health_harms': ['coughing', 'wheezing', 'asthma']}

rnt_net = rnt.GetRedditNetwork(reddit_dataset=data,
                               edge_type='directed',
                               text_attribute=attribute_dictionary)

print(rnt_net)

print('NetworkX graph object:')
print(rnt_net.graph)

print('Edge list variables in the data set:')
print(rnt_net.edge_list.columns)

# Node list
print(rnt_net.node_list.columns)

print('Dictionary of adjacency matrices:')
print(rnt_net.adjacency.keys())

rnt_net.write_data(file_name='e_cigarette_data',
                   adjacency=True)
```

Console output

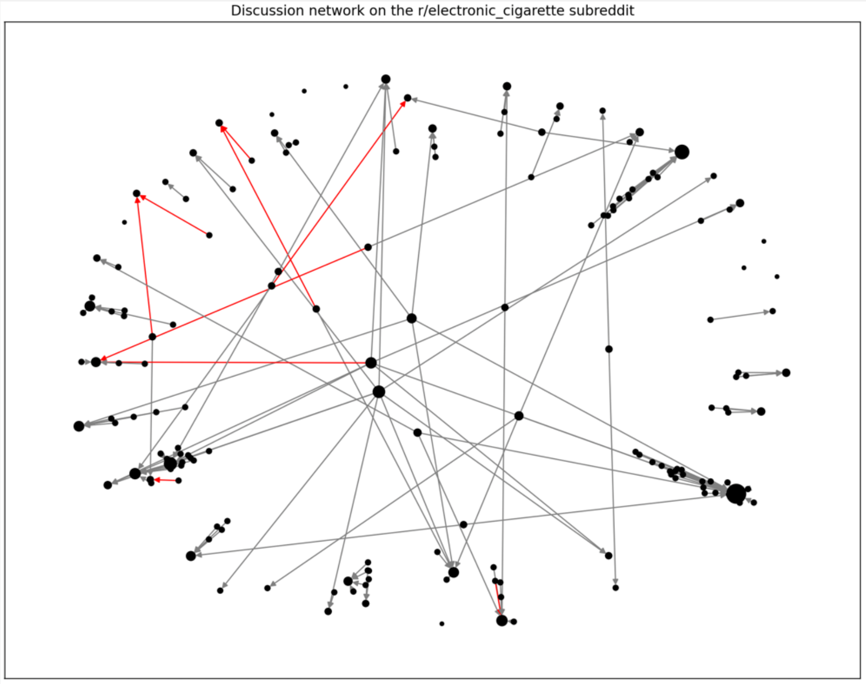
```
Reddit network object:
Number of nodes: 162
Number of edges: 149

NetworkX graph object:
DiGraph with 162 nodes and 149 edges

Edge list variables in the data set:
['source', 'target', 'weight', 'subreddit', 'text_attribute_addiction',
'text_attribute_health_harms']

Node list variables in the data set:
['node', 'degree', 'in_degree', 'out_degree', 'node_subreddits']

Dictionary of adjacency matrices:
dict_keys(['weighted_adj_matrix', 'unweighted_adj_matrix',
'text_attribute_addiction_matrix', 'text_attribute_health_harms_matrix'])

Writing weighted_adj_matrix_e_cigarette_data.csv to file.
Writing unweighted_adj_matrix_e_cigarette_data.csv to file.
Writing text_attribute_addiction_matrix_e_cigarette_data.csv to file.
Writing text_attribute_health_harms_matrix_e_cigarette_data.csv to file.
```

Code snippet 3. Visualize Reddit network data using *rnt*.

Console input

```
G = rnt_net.graph

node_size_list = [(G.degree(node) + 1) * 10 for node in G.nodes]

edge_color_list = []

for edge in G.edges.data():
    if edge[2]['text_attribute_addiction'] == 'True':
        edge_color_list.append('red')

    elif edge[2]['text_attribute_health_harms'] == 'True':
        edge_color_list.append('red')

    else:
        edge_color_list.append('grey')

plot_title = 'Discussion network on the r/electronic_cigarette subreddit'

rnt.single_network_plot(network=rnt_net,
                        title=plot_title,
                        pos='spring_layout',
                        node_size=node_size_list,
                        edge_color=edge_color_list)
```

Console output



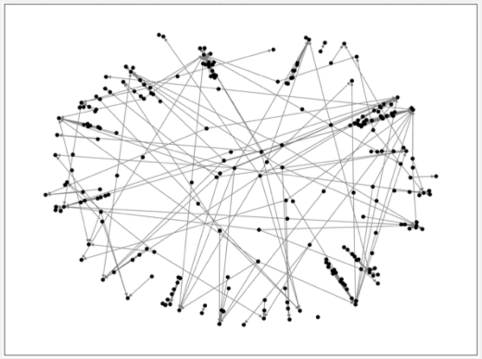
Code snippet 4. Using *rnt* with other data analysis packages.

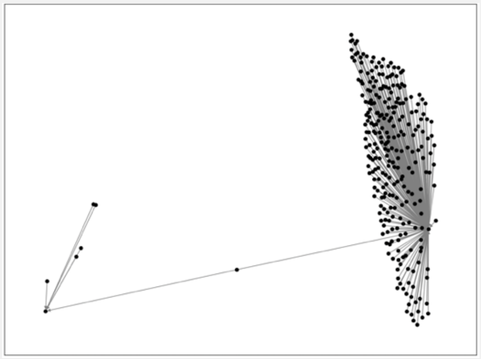
Console input

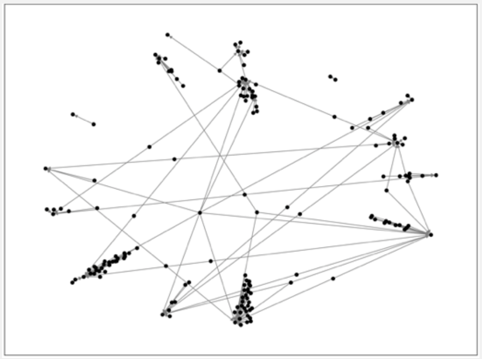
```
import pandas

keywords = ['vaping', 'vape', 'e-cig', 'ecig']
data_2 = rnt.GetRedditData(search_term=keywords, size=500)

print(f'Basic descriptive statistics of the data:')
print(f'Number of unique subreddits: {len(data_2.subreddit.unique())}')
print(f'Number of unique Redditors: {len(data_2.author.unique())}')

print(f'Top 3 subreddits:')
print(data_2.subreddit.value_counts().head(3))

print('Example thread-level data for the top 3 subreddits:')
print(thread_data[['author', 'subreddit', 'num_unique_responders', 'mean_response_time']].head(3))

# subset the data to only the top 3 subreddits
top_three_subs = data_2.subreddit.value_counts().head(3).index
top_subreddits_df = data_2.loc[data_2.subreddit.apply(lambda x: x in top_three_subs)]

# generate subreddit- and thread-level data
thread_data = rnt.reddit_thread_statistics(top_subreddits_df)
subreddit_data = rnt.subreddit_statistics(top_subreddits_df)

print('Example subreddit-level data for the top 3 subreddits:')
print(subreddit_data[['subreddit', 'num_submissions', 'num_comments', 'number_graph_nodes', 'density']])

for subreddit in top_three_subs:
    subreddit_net = rnt.GetRedditNetwork(top_subreddits_df.loc[top_subreddits_df.subreddit == subreddit])
    rnt.single_network_plot(subreddit_net)
    print(f'r/{subreddit} edge list data:')
    print(subreddit_net.edge_list.loc[:, subreddit_net.edge_list.columns != 'subreddit'].head(3))
```

Console output

```
Basic descriptive statistics of the data:
Number of unique subreddits: 250
Number of unique Redditors: 2632

Top 3 subreddits:
Vaping       554
Teachers     337
vaporents    332

Example thread-level data for the top 3 subreddits:
  author  subreddit  num_unique_responders  mean_response_time
0  CUDBD     Vaping                     26        33716.333333
1  ZIZNW  vaporents                     35        45642.931818
2  YUCYF     Vaping                      4         8622.400000

Example subreddit-level data for the top 3 subreddits:
   subreddit  num_submissions  num_comments  number_graph_nodes   density
0     Vaping               45           509                 245  0.004567
1  vaporents               17           315                 163  0.006741
2   Teachers                2           335                 272  0.003676
```

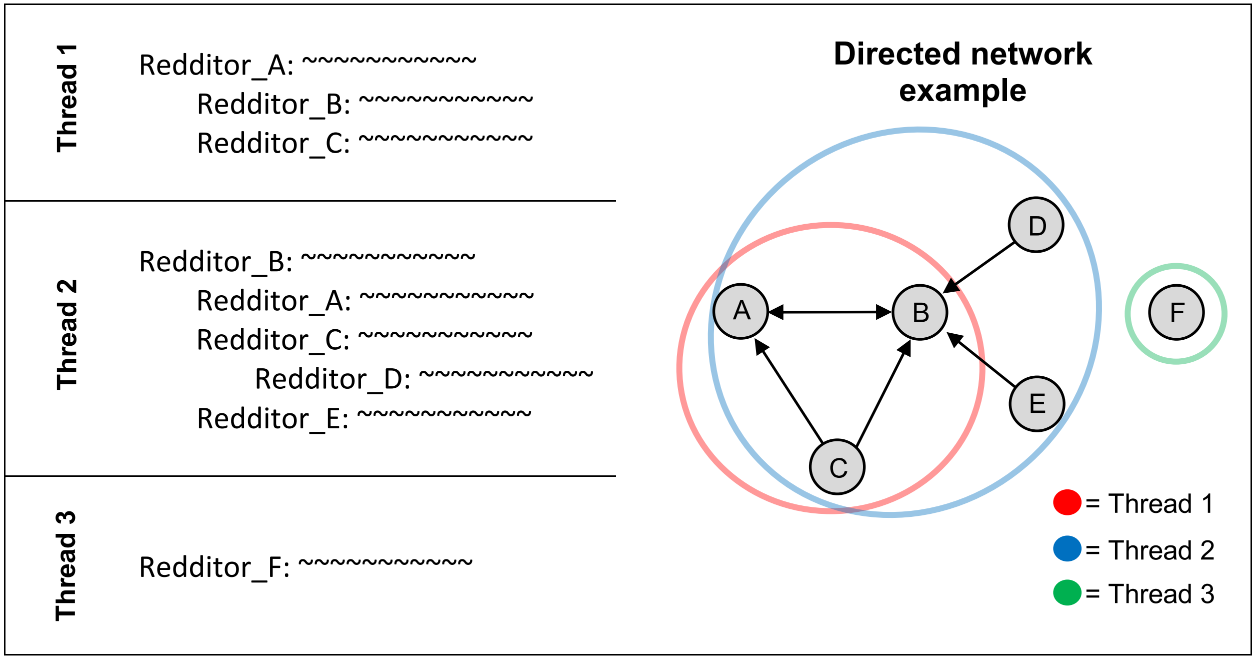






```
r/Vaping edge list data:          r/Teachers edge list data:          r/vaporents edge list data:
  source target  weight             source target  weight               source target  weight
0  RIOVE  ANXZL       1           0  MRGPR  CPULK       2             0  QHCRG  XUSHS       2
1  RIOVE  UUICD       3           1  XLCQD  CPULK       1             1  WYOJC  XUSHS       2
2  ZDRWV  HUOKI       2           2  XLCQD  BEKML       1             2  PYNPF  XUSHS       1
```

Figure 1. *rnt* network structure.


Thread 1
Redditor_A: ~~~~~~~~~~~~
Redditor_B: ~~~~~~~~~~~~
Redditor_C: ~~~~~~~~~~~~
Thread 2
Redditor_B: ~~~~~~~~~~~~
Redditor_A: ~~~~~~~~~~~~
Redditor_C: ~~~~~~~~~~~~
Redditor_D: ~~~~~~~~~~~~
Redditor_E: ~~~~~~~~~~~~
Thread 3
Redditor_F: ~~~~~~~~~~~~
Directed network example
A
B
C
D
E
F
● = Thread 1
● = Thread 2
● = Thread 3