The Journal of Open Source Software

BiRank: Fast and Flexible Ranking on Bipartite Networks with R and Python

Kai-Cheng Yang[1], **Brian Aronson**[2], **and Yong-Yeol Ahn**[1]

1 Luddy School of Informatics, Computing, and Engineering, Indiana University, Bloomington, IN **2** Department of Sociology, Indiana University, Bloomington, IN

DOI: 10.21105/joss.02315

Software

- Review
- Repository
- Archive

Editor: Vincent Knight

Reviewers:

- @gvegayon
- @nikoleta-v3

Submitted: 18 February 2020
Published: 10 July 2020

Summary

Bipartite (two-mode) networks are ubiquitous. Common examples include networks of collaboration between scientists and their shared papers, networks of affiliation between corporate directors and board members, networks of patients and their doctors, and networks of competition between companies and their shared consumers. Bipartite networks are commonly reduced to unipartite networks for further analysis, such as calculating node centrality (e.g. PageRank, see Figure 1(c)). However, one-mode projections often destroy important structural information (Lehmann, Schwartz, & Hansen, 2008) and can lead to imprecise network measurements. Moreover, there are numerous ways to obtain unipartite networks from a bipartite network, each of which has different characteristics and idiosyncrasies (Bass et al., 2013).

To overcome the issues of one-mode projection, we present BiRank, an R and Python package that performs PageRank on bipartite networks directly. The BiRank package contains several ranking algorithms that generalize PageRank to bipartite networks by propagating the probability mass (or importance scores) across two sides of the networks repeatedly using the following equations:

$$\mathbf{T} = \alpha S_T \mathbf{B} + (1-\alpha)\mathbf{T}^0$$
$$\mathbf{B} = \beta S_B \mathbf{T} + (1-\beta)\mathbf{B}^0$$

until they converge (see Figure 1(a)), where $\mathbf{T}, \mathbf{B}$ are the ranking values for the top and bottom nodes, elements in $\mathbf{T}^0$ and $\mathbf{B}^0$ are set to $1/|\mathbf{T}|$ and $1/|\mathbf{B}|$ by default, α and β are damping factors and set to 0.85 by default, S_T, S_B are the transition matrices. Unlike the one-mode projected PageRank, BiRank algorithms generate the ranking values for nodes from both sides simultaneously and take account of the full network topology without any information loss.

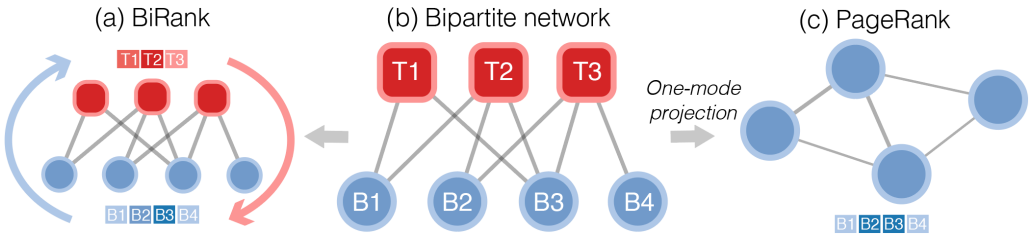

Figure 1: (a) BiRank algorithms perform the ranking process on the bipartite networks directly and generate the ranking values for the top and bottom nodes simultaneously. (b) A bipartite network with three top nodes and four bottom nodes. (c) After the one-mode projection, a unipartite network of the bottom nodes is generated. PageRank can be performed to generate the ranking values of the bottom nodes.

Our package implements the most notable and straightforward operationalizations of biparitite PageRanks including HITS (Kleinberg, 1999; Liao, Xiao, Cimini, & Medo, 2014), CoHITS (Deng, Lyu, & King, 2009), BGRM (Rui, Li, Li, Ma, & Yu, 2007), and Birank (He, Gao, Kan, & Wang, 2016). The algorithms mainly differ in the way they normalize node ranks in the iterations (see Table 1).

Table 1: A summary of transition matrices used in different BiRank algorithms. K_T and K_B are diagonal matrices with generalized degrees (sum of the edge weights) on the diagonal, i.e. $(K_T)_{ii} = \sum_j w_{ij}$ and $(K_B)_{jj} = \sum_i w_{ij}$. w_{ij} is the element on row i and column j of the bipartite network adjacency matrix $W^{|T|\times|B|}$.

Transition matrix	S_B	S_T
HITS	$W^\top$	W
Co-HITS	$W^\top K_T^{-1}$	$W K_B^{-1}$
BGRM	$K_B^{-1} W^\top K_T^{-1}$	$K_T^{-1} W K_B^{-1}$
Birank	$K_B^{-1/2} W^\top K_T^{-1/2}$	$K_T^{-1/2} W K_B^{-1/2}$

Our guiding philosophy is to make the package as flexible as possible, given the diverse array of problems and data formats that are used in network analysis, while achieving good performance. We therefore provide a number of convenience options for incorporating edge weights into rank estimations, estimating ranks on different types of input (edge lists, dense matrices, and sparse matrices), multiple file formats (as vectors, lists, or data frames), and for estimating PageRank on the one-mode projection of a network. Moreover, this implementation uses efficient data storage and algorithms to ensure good performance and scalability. For example, regardless of the algorithm of choice, it takes less than 10 seconds and less than 1GB of RAM to estimate ranks on a bipartite network containing half a million top nodes, more than two million bottom nodes, and about three million edges on a machine with 16 AMD EPYC 7000 series 2.5 GHz processors.

As a demonstration, we apply HITS, CoHITS, and one-mode projected PageRank to the Marvel Universe collaboration network (Alberich, Miro-Julia, & Rosselló, 2002). The Marvel Universe collaboration network comprises a network of affiliation with ties between every Marvel comic book (n = 12,849) and every character (n = 6,444) who appeared in those books. To give a sense of this network's structure, Figure 2 illustrates a small sociogram of characters within ten comic books of this dataset.

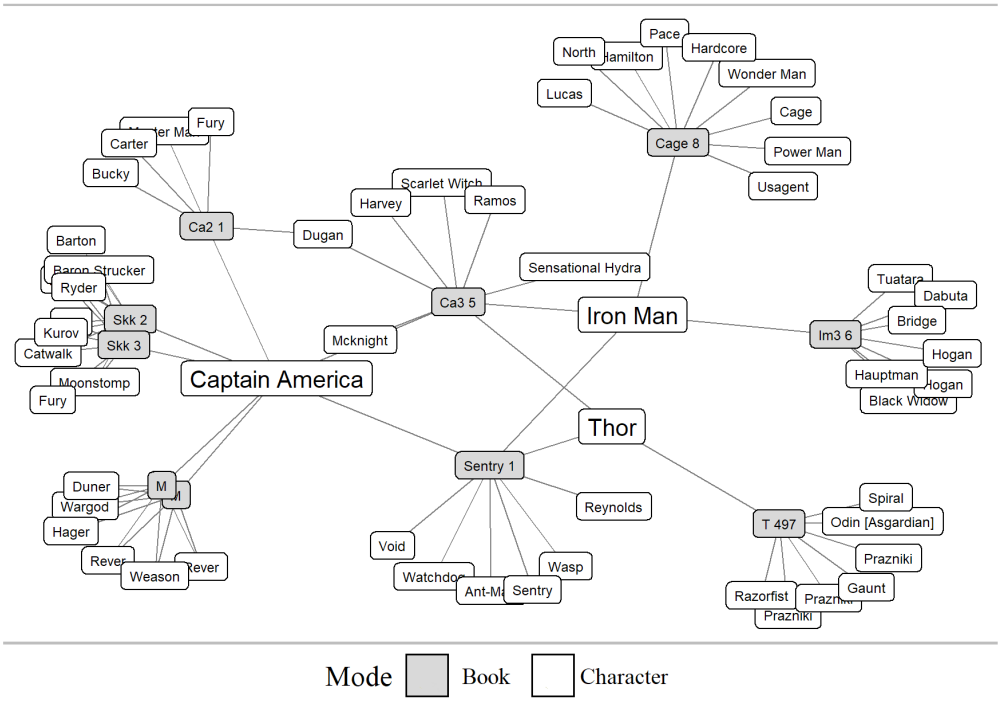

Figure 2: Sociogram of character-book ties within 10 comic books of the Marvel Universe collaboration network.

Table 2 presents the five characters with the highest ranking values from each algorithm. Results are similar, with Captain America and Iron Man occurring in all three ranking algorithms. However, discrepancies arise from differences in the underlying ranking algorithms and how they interact with the network's structure. PageRank on the one mode projection first converts comic-character ties to character-character ties. Without information about the structure of characters-comic ties, PageRank mainly prioritizes nodes with a large number of transitive ties in the original network. For example, Wolverine has a higher PageRank than the Thing but Wolverine appears in much fewer comic books than the Thing. Instead, Wolverine's high PageRank is a result of his co-presence in comic books with large numbers of other characters. In contrast, the Thing tends to repeatedly appear in central comic books with other central characters in the Marvel universe, hence the Thing has a high CoHITS rank but a lower PageRank than Wolverine.

Table 2: Top five characters in the Marvel Universe collaboration network ranked by HITS, CoHITS and PageRank with one-mode projection.

Rank	HITS	CoHITS	Projection+PageRank
1st	Captain America	Spider-man	Captain America
2nd	Iron man	Captain America	Spider-man
3rd	Thing	Iron man	Iron man
4th	Human torch	Hulk	Wolverine
5th	Mr. fantastic	Thing	Thor

Differences between how HITS and CoHITS estimate ranks on the Marvel Universe collaboration network are more complicated. CoHITS normalizes the transition matrix by the outdegree of the source nodes, and therefore places somewhat less value on connections from highly connected characters and from highly connected comic books than HITS. As a result, CoHITS

tends to assign higher ranks to characters who are connected to a more diverse array of comic books than does HITS. This difference is best illustrated by the inclusion of Mr. Fantastic in HITS' top-ranked characters and the inclusion of Spider Man in CoHITS' top-ranked characters: Spider Man appears in nearly twice as many comic books as Mr. Fantastic and collaborates with a significantly wider cast of characters than Mr. Fantastic; however, Mr. Fantastic tends to appear in highly central comic books with large character casts. It is open to interpretation as to which measure of centrality is better, but in many applications, we tend to prefer CoHITS over HITS as CoHITS ranks are less influenced by the presence of outliers with extreme degrees (Aronson, Yang, Odabas, Ahn, & Perry, 2020).

It is also worth mentioning that assigning different edge weights to the network can significantly affect ranking results. Our package offers flexibility by allowing different combinations of algorithms and edge weights. We leave the choice to the users' discretion.

Despite the ubiquity of bitpartite networks, bipartite PageRank algorithms are missing from the popular network packages, and our package serves to close this gap. Our target audience includes researchers and data scientists who deal with bipartite networks. To improve the accessibility, both R (birankr) and Python (birankpy) versions of the package are available. The documentation of BiRank consists of manual pages for its method functions, example usages, and unit tests.

Acknowledgement

The authors acknowledge support from National Institute on Drug Abuse (grant R01 DA039928).

References

Alberich, R., Miro-Julia, J., & Rosselló, F. (2002). Marvel universe looks almost like a real social network. *arXiv preprint cond-mat/0202174*.

Aronson, B., Yang, K.-C., Odabas, M., Ahn, Y. Y., & Perry, B. L. (2020, June). Comparing measures of centrality in bipartite social networks: A study of drug seeking for opioid analgesics. SocArXiv. doi:10.31235/osf.io/hazvs

Bass, J. I. F., Diallo, A., Nelson, J., Soto, J. M., Myers, C. L., & Walhout, A. J. (2013). Using networks to measure similarity between genes: Association index selection. *Nature methods*, *10*(12), 1169. doi:10.1038/nmeth.2728

Deng, H., Lyu, M. R., & King, I. (2009). A generalized co-hits algorithm and its application to bipartite graphs. In *Proceedings of the 15th acm sigkdd international conference on knowledge discovery and data mining* (pp. 239–248). New York, NY, USA: ACM. doi:10.1145/1557019.1557051

He, X., Gao, M., Kan, M.-Y., & Wang, D. (2016). Birank: Towards ranking on bipartite graphs. *IEEE Transactions on Knowledge and Data Engineering*, *29*(1), 57–71. doi:10.1109/TKDE.2016.2611584

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM*, *46*(5), 604–632. doi:10.1145/324133.324140

Lehmann, S., Schwartz, M., & Hansen, L. K. (2008). Biclique communities. *Physical Review E*, *78*(1), 016108. doi:10.1103/PhysRevE.78.016108

Liao, H., Xiao, R., Cimini, G., & Medo, M. (2014). Network-driven reputation in online scientific communities. *PLoS one*, *9*(12), e112022. doi:10.1371/journal.pone.0112022

Rui, X., Li, M., Li, Z., Ma, W.-Y., & Yu, N. (2007). Bipartite graph reinforcement model for web image annotation. In *Proceedings of the 15th acm international conference on multimedia* (pp. 585–594). New York, NY, USA: ACM. doi:10.1145/1291233.1291378