Community Detection on the Reddit Hyperlink Dataset

Alexander Rundle
Department of Computer Science
University of Exeter, Exeter, UK

Internal Supervision:
[redacted]
University of Exeter, Exeter, UK

Internal Supervision:
[redacted]
University of Exeter, Exeter, UK

***Abstract*—WRITE ABSTRACT AT END OF WRITING REPORT**

I. Introduction: Contextualization and Motivation

In this report, I detail the use of community detection algorithms when applied to the Reddit Hyperlink Dataset [1], a dataset of connections between established communities on the popular forum-based website Reddit. Doing this, I apply and compare three different community detection algorithms: the Louvain algorithm [2], the Girvan-Newman algorithm [3] and the Label Propagation algorithm [4].

In this Section I contextualise this report, discussing the chosen dataset, as well as providing the motivation for the algorithms used. Following this in Section II, I detail pre-processing that is applied to the dataset to ensure that it works with the chosen algorithms; I follow this with explanations of each of the three algorithms alongside details of their implementation in Python 3.9 [5]. I then discuss in Section III the results of the community detection on the Reddit Hyperlink Dataset [1], as well as comparing the efficacy, pros, and cons of each algorithm that has been used. Finally in Section IV, I conclude this paper with some remarks about future research avenues and an overview of the report.

A. The Reddit Hyperlink Dataset

For this mini-project I have chosen to use the Reddit Hyperlink Dataset [1], hosted on the Stanford Large Network Dataset Collection [6]. The Reddit Hyperlink Dataset was created by Kumar et al. [1] to study cases of 'intercommunity conflict', where members of one Reddit community (called a 'Subreddit') collectively mobilise to participate in or attack another community [1].

The dataset itself consists of a Tab-Separated Values (TSV) file containing 40 months of Reddit comments and posts extracted between January 2014 and April 2017, in which a hyperlink to another subreddit was present. As such, each entry in the dataset represents a directed edge between two subreddit nodes and allows one to study communities that may exist between various subreddits. The dataset also provides the Post ID, the timestamp of the post, a label indicating explicit sentiment towards the target subreddit and a vector representing the text properties of the source post [1].

II. Methods: Description and Implementation

Before any community detection algorithms can be applied to the Reddit Hyperlink Dataset, some pre-processing must be carried out to ensure that it works with the NetworkX package [7]. First, the original TSV file is downloaded from the Stanford Large Network Dataset Collection and loaded into a temporary Pandas DataFrame [8], [9]. The temporary DataFrame is then saved as a Comma-Separated Values (CSV) file to allow for easier manipulation; following this, the CSV file is loaded into a new DataFrame in which only the `SOURCE_SUBREDDIT` and `TARGET_SUBREDDIT` fields are extracted.

As each row in the CSV file represents any post or comment containing a hyperlink from the source subreddit to the target subreddit, rows can repeat source and target subreddits. To convert this DataFrame into an edgelist, the occurrences of each row are counted and then placed into a `WEIGHT` field for each edge. For example, if a row containing the source 'destinythegame' and target 'gaming' subreddits respectively appears 5 times within the CSV file, then the weight for the edge from 'destinythegame' to 'gaming' would be 5. The weighted edgelist is subsequently saved as a separate CSV file, which is provided within the submission.

A. Louvain Algorithm

Introduced in 2008 by Blondel et al. [2], the Louvain algorithm finds partitions of high modularity in large networks by unfolding the network into a complete hierarchical structure [2]. The algorithm itself is divided into two phases, which are repeated iteratively; initially in phase one, each node is assigned its own community so that there are as many communities as are nodes in the network. For each node i and its neighbours j of i, the gain in modularity that might occur if i was moved from its own community to the community of j is evaluated; node i is subsequently placed into the community for which the modularity gain is a positive maximum [2]. This is repeated for every node, repeatedly until a local maxima of modularity is achieved. In the second phase of the algorithm, a new network is built whose nodes are now the communities found during phase one. The weights on edges between the new nodes are found as the sum of the weights between nodes in the corresponding two communities [10]. Once this new network is built, the first phase of the algorithm is then reapplied and iterated; in this case, a 'pass' is denoted as a

combination of the two phases [2]. The passes are then iterated until no changes are made and a global maxima of modularity is attained.

To apply this algorithm to the Reddit Hyperlink Dataset [1], I use the `python-louvain` package [11] for Python 3.9 [5]. The `community.best_partition()` function is used here and returns a dictionary containing each node and the numerical label of the community that it belongs to. I follow this by unfolding the dictionary into a DataFrame [8] which can be used to plot each nodes based on community.

B. Girvan-Newman Algorithm

Preceding the Louvain Algorithm [2], in 2002 Girvan & Newman [3] introduced a community detection algorithm which aimed to sidestep the problems with previous hierarchical clustering methods, where one first calculates a weight for each pair of vertices in the network that denotes pairs that might be the most central within a community, and progressively adds edges between pairs in order of weight [3]. Instead, Girvan & Newman [3] propose an algorithm that progressively removes the *least* central edges from the original graph, in order of edge betweenness centrality. The betweenness centrality of a node was first defined by Freeman [12] as the number of nodes which run through a node i; it is the measure of the influence a given node has over the flow of information between other nodes.

Girvan & Newman generalise the node betweenness centrality to edges, such that the edge betweenness is the number of shortest paths of nodes that run along said edge [3]. If there is more than one shortest path between a pair of nodes, each path is given an equal weight so that they all sum to 1. The algorithm is simply stated in the following steps:

1) Calculate the edge betweenness for all edges in the network.
2) Remove the edge with the highest betweenness.
3) Recalculate the betweenness for all edges affected by the removal.
4) Repeat step 2 until no edges remain.

To ensure the algorithm is performant, they utilise Newman's fast betweenness algorithm [13], which calculates betweenness for m edges in a graph of n nodes in time $O(mn)$. This calculation is repeated once for the removal of each edge, so the Girvan-Newman algorithm runs in the worst-case time of $O(m^2n)$.

C. Label Propagation Algorithm

[14]

III. Experiments and Results

IV. Conclusions

A. Future Work

References

[1] S. Kumar, W. L. Hamilton, J. Leskovec, and D. Jurafsky, "Community interaction and conflict on the web," in *Proceedings of the 2018 World Wide Web Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2018, pp. 933–943.

[2] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of statistical mechanics: theory and experiment*, vol. 2008, no. 10, p. P10008, 2008.

[3] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[4] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical review E*, vol. 76, no. 3, p. 036106, 2007.

[5] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.

[6] J. Leskovec and A. Krevl, "SNAP Datasets: Stanford large network dataset collection," http://snap.stanford.edu/data, Jun. 2014.

[7] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring network structure, dynamics, and function using networkx," in *Proceedings of the 7th Python in Science Conference*, G. Varoquaux, T. Vaught, and J. Millman, Eds., Pasadena, CA USA, 2008, pp. 11 – 15.

[8] The Pandas Development Team, "pandas-dev/pandas: Pandas," Feb. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3509134

[9] Wes McKinney, "Data Structures for Statistical Computing in Python," in *Proceedings of the 9th Python in Science Conference*, Stéfan van der Walt and Jarrod Millman, Eds., 2010, pp. 56 – 61.

[10] A. Arenas, J. Duch, A. Fernández, and S. Gómez, "Size reduction of complex networks preserving modularity," *New Journal of Physics*, vol. 9, no. 6, p. 176, 2007.

[11] T. Aynaud, "python-louvain x.y: Louvain algorithm for community detection," `https://github.com/taynaud/python-louvain`, 2020.

[12] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, pp. 35–41, 1977.

[13] M. E. Newman, "Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality," *Physical review E*, vol. 64, no. 1, p. 016132, 2001.

[14] F. Parés, D. G. Gasulla, A. Vilalta, J. Moreno, E. Ayguadé, J. Labarta, U. Cortés, and T. Suzumura, "Fluid communities: A competitive, scalable and diverse community detection algorithm," in *Complex Networks & Their Applications VI: Proceedings of Complex Networks 2017 (The Sixth International Conference on Complex Networks and Their Applications)*. Springer, 2018, pp. 229–240.