

Hello I'm a title

Network Tour of Data Science Report, January 2019

Isabela Constantin, Adélie Garin, Celia Hacker, Michael Spieler

I. INTRODUCTION

Using the Wikipedia data set that is now open source, we would like to understand how the number of views on one page spreads in the linked pages when some unexpected event happens. We chose to study specifically the pages of three famous people who died in the past few years and we are interested in the number of views in the pages that are related to the main page of the person who died. We will consider two times for the number of views: the day before the person dies and the day of the death.

For each person who died, we build a specific graph around their wikipedia page by taking wikipedia pages as nodes and links from one page to another as directed edges. We will use both the directed graph, in which one page which is linked to the other leads to only one direction for the concerned edge, and the undirected graph, for which there exists an edge if and only if there is one page linked to the other. The full construction of the three graphs is described in Section II. For each node i (corresponding to a wikipedia page) of a graph, we are interested in the following quotient:

$$Q_i := \frac{\text{number of views of the page on the day of death}}{\text{number of views of the page the day before}}.$$

The higher this value is, the bigger difference there is between the number of views of the node i on the day before the person died and the day of their death.

The question we ask ourselves is the following: does the number of views on the pages related to the one of the famous person depend on the number of views of the page of the person itself?

To answer this question, we will study the quotient Q_i for each node i of the graph and analyse the different values Q_i as a signal on the graph. We build a general pipeline to analyse the three graphs in the same matter, to be able to compare them. Our general pipeline is the following:



We first start by acquiring the data and building the three graphs around the people we chose, in Section II. Then, we explore the graphs and try to extract their general properties in Section III. In Section IV, we exploit the data, trying to test our hypothesis and we finally draw conclusions in Section V.

II. DATA ACQUISITION

We constructed the graphs around the selected Wikipedia articles corresponding to each of the following people: Stephen Hawking [?], Stan Lee [?] and Alan Rickman [?]. To make it easier for the following description, we will say that a node or page *links in* another one if there is a link from the former to the latter, and *links out* if the link is from the latter to the former. We grew the graphs around the main pages (Stephen Hawking, Stan Lee, Alan Rickman) by selecting the pages linked out from the main ones (step 1), then we added pages linked out from the pages at step 1 (step 2), and so on. At each step i , we hence added pages that were linked out from the previous pages of step $i - 1$. Essentially, we crawled the pages by a slightly modified version of Bread First Search.

A. Sampling

It would be too much data to add all the linked out pages at each step. To reduce the amount of data, we subsampled by making several choices of parameters, described as follow. First, we randomly subsampled the nodes by giving a higher probability of being added in the graph to the pages which link is higher (in order of appearance) in previous pages. This is because pages that are stated early in a Wikipedia page are more likely to be really related to the subject of the page. We hence set up two percentage parameters θ_1 and θ_2 . Let us take a page at step i . The first θ_1 percent of links from that page constitute for θ_2 percent of the edges linked out from this page at step $i + 1$. Our next choice of parameter is the number of neighbors for the main page, i.e. the number of pages that will be taken at step 1. We call this parameter N_{step1} . We also assign a minimum of popularity parameter, m_{pop} , which is the minimum number of out-links on a page for the page to be considered as a possible node. The last parameter that we can set up is the maximal number of nodes N_{max} .

The parameters are summed up in the following table.

TABLE I
PARAMETERS

	Use	Name in code	Value
θ_1	Top percentage of links	percentage_1	0.2
θ_2	Percentage of subsampled links	percentage_1_subsample	0.5
N_{step1}	Number of pages taken at step 1	n_links_hop1	150
m_{pop}	Minimum popularity	min_popularity	30
N_{max}	Max number of nodes	max_nodes	2701

We set up the values of the parameters for the pages that are important and related to the subject to have a higher

probability to be in the graphs. We chose them such that the top 20 percent of the links account for 50 percent of the sampled links (parameters θ_1 and θ_2). At step 1, we added 150 links (parameter N_{step1}). To be considered as a possible node, a page needed at least 30 out-links (parameter m_{pop}). Last but not least, the maximum number of nodes in the graph, $N_{max} = 2701$, was selected at random between $150 \times 3 \times 3$ and $150 \times 3 \times 3 \times 3$ to make sure that we would have at least 5 steps in the process. Indeed, at each step we added on average 3 links from each node, and we started with 150 nodes after step 1.

The resulting graphs are directed, unweighted and connected.

B. Building the views signal

We found the number of views and extracted them at each page for the day before the death of the person and the day of the death. For each page i we computed the quotient

$$Q_i := \frac{\text{number of views of the page on the day of death}}{\text{number of views of the page the day before}},$$

and we assign this value to the corresponding node of the graph. Taking this quotient is a sort of normalization of the number of views. The higher this value is, the more the number of views on the day of the death is big compared to the number of views of the day before. In Section IV, we consider these values as a signal on the graph and we study their diffusion.

III. DATA EXPLORATION

In this section, we analyse basic properties of the graphs. One point can be made with this section: an analysis of the same signal on those three graphs would make sense, because the three graphs are very similar.

Let us first summarize the number of views of the pages for Stephen Hawking, Stan Lee and Alan Rickman.

TABLE II
NUMBER OF VIEWS

Views	Day of death	Day before	Quotient
Stephen Hawking	7126234	15781	451.57
Stan Lee	2404717	6929	347.05
Alan Rickman	3394010	3664	926.31

We can clearly see that the number of views grew a lot on the day of the death of each of them. The quotients are of the same order.

We also computed the graphs of the number of views several days before and after the death just for those three pages.

A. Basic properties

We start by analysing some basic properties of each graph, such as the number of nodes, the number of edges, the diameter of the (undirected) graph, and the average clustering coefficient (ACC), number of triangles and global clustering

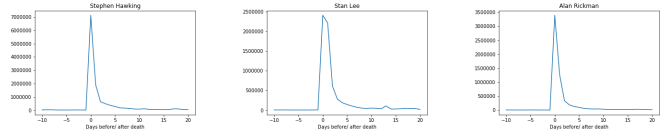


Fig. 1. Number of views for Stephen Hawking, Stan Lee and Alan Rickman. For each person, we can see that there is a high peak of views on the days of the death, then the number of views quickly decays to the baseline.

TABLE III
BASIC PROPERTIES OF THE GRAPHS

Properties	Stephen Hawking	Stan Lee	Alan Rickman
#nodes	2701	2701	2701
#edges	26921	33469	28012
Diameter	7	7	8
ACC	0.138	0.131	0.123
#triangles	41973	60096	38076
GCC	0.000335	0.000379	0.000251

coefficient (GCC) with the tools available in Networkx. The results are stated in the following table.

The number of nodes are the same because they were set to be the same. The number of edges are of the same order, as well as the diameters and average and global clustering coefficients. This comes partially from the way the graphs were built in the same way and partially from the fact that we built them around very similar type of pages, famous people. The number of triangles is slightly higher for Stan Lee, most likely because there are more edges in the graph. Both the average clustering coefficient and the global clustering coefficient are very low. This comes from the construction of the graphs, they were built around one node, adding edges one by one. Hence, very few triangles are built on the way.

B. Adjacency matrices

The following figure shows the adjacency matrices of the undirected graphs.

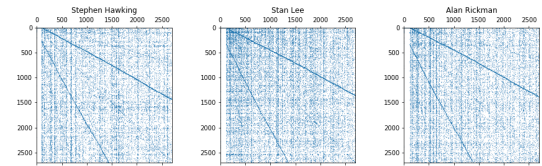


Fig. 2. Undirected adjacency matrices for Stephen Hawking, Stan Lee and Alan Rickman's graphs.

The three matrices are very similar. The straight lines "in diagonal" that appear in the three of them come from the construction of the graphs. The graphs were built by "chains", we start with one node and add many edges out of it. This will create a lot of entries nearby in the adjacency matrix. Starting from node i , a node j is added, linked out to i . Then a node k is added, linked out to j , and so on, with i , j , and k quite close to each other. This gives those lines structure in the matrices.

C. Degree Distribution

Here we study the degree distributions of each graph. We computed degree distributions with logarithmic scale for in-degrees, out-degrees and undirected case. The two most relevant, based on the graph construction, are the out-degree and the undirected case, that we show on the following figures.

Fig. 3. Degree distribution of the graphs for Stephen Hawking, Stan Lee and Alan Rickman in logarithmic scale for the undirected case (top line) and out-degree (bottom line).

D. Strongly Connected Components

A directed graph is said to be strongly connected if there is a directed path connecting any two of its vertices. Our graphs are clearly connected in the undirected sense by the way we have constructed them. All of them are not strongly connected, so we computed the number of strongly connected components. Stephen Hawking's graph has 51 strongly connected components, both Stan Lee and Alan Rickman's graphs have 55 strongly connected components.

IV. DATA EXPLOITATION

We now come to the most important part of our analysis, which is the following: we consider the number of views as a signal on the graph. We would like to see how it behaves. In order to make a sensible comparison of number of views and how these number of views evolve for each graph we use the quotient of the number of views before and at the day of the event. The following figure shows the signal on the graphs.

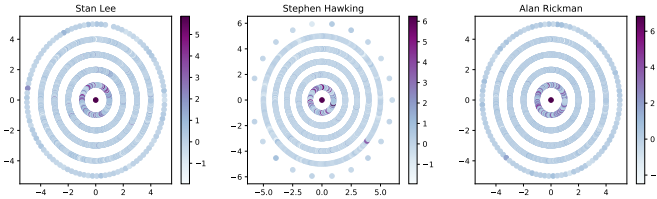


Fig. 4. Only the nodes of the graphs are represented here. The central node of each graph stands for Stephen Hawking, Stan Lee and Alan Rickman respectively. One can see very clearly that the quotient signal is much higher in the center, near the main person, and that it decreases quite fast when going away from the center.

FURTHER ANALYSIS OF THIS

V. CONCLUSION

Note that if we had the computational power to build a graph with all the pages involved by all the events that we considered together, it would have been much more interesting to see how the number of views evolves locally on the graph. By doing the methods presented in this report, we can only do a “zoom-in” on a specific part of this big graph and analyse it independently of the rest.

IF YOU DIE; PEOPLE WILL SHOW INTEREST IN YOU:
BUT ONLY MOMENTANELY SO DONT DIE

REFERENCES

[?] [?] [?]

REFERENCES

- [1] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.
- [2] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30(3):83–98, 2013.
- [3] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.
- [4] Wikipedia. Alan rickman. https://en.wikipedia.org/wiki/Alan_Rickman.
- [5] Wikipedia. Stan lee. https://en.wikipedia.org/wiki/Stan_Lee.
- [6] Wikipedia. Stephen hawking. https://en.wikipedia.org/wiki/Stephen_Hawking.