Information retrieval - Assignment 2

PageRank to rank web pages and graph analysis

Group members:

- Laura Gotra
- Mirko Gaslini
- Martina Ceccon

Contents

1 Introduction

The topic we choose for this second assignment is PageRank's topic. The goal of this report is to show how to implement a PageRank's algorithm to rank web pages and understand the underlying structure, logic behind some algorithms that rank web pages on the internet. Moreover it will be shown a detailed network's analysis, in particular some comparisons between centrality's measures and PageRank's score. All the code and results can be view on this **link.**

2 Page Rank

The goal of this section is to introduce the PageRank's concepts, techniques and provide some general information about what the PageRank is and what is it's utility. *"PageRank (PR) is Google's main method of ranking web pages for placement on a search engine results page (SERP). PageRank refers to the system and the algorithmic method that Google uses to rank pages as well as the numerical value assigned to pages as a score"* [8].

2.1 How PageRanks works

PageRank works by counting the number and quality of links to a page to determine how much one website is important. Usually more important a page is (high PageRank score) more important will be all the pages that this page is connected.
"The connections between a web pages can be represented as a web graph. Every node represents a page and a directed arrow from one node to another means that there is a link between two pages." [6].
PageRank can be computed by an iterative algorithm. In the PageRank algorithm the damping factor d is significant, this parameter state how much time one random web surfer follows hyperlink structure than teleporting in a random page. Using this damping factor d is useful to prevent sink nodes from "absorbing" the PageRanks of those pages connected to the sinks. The damping factor is subtracted from 1 and also this term is then added to the product of the damping factor and the sum of the incoming PageRank scores. Normally the value of damping factor is 0.85 (in this

report it will show the results using different values of damping factor).
The values of PageRank can be obtained by solving the subsequent set of linear equations

$$\begin{pmatrix} p1 \\ p2 \\ \dots \\ pn \end{pmatrix} = d \begin{bmatrix} A_{11}A_{12}A_{13}\dots A_{1n} \\ A_{21}A_{22}A_{23}\dots A_{2n} \\ \dots\dots\dots\dots\dots\dots \\ A_{n1}A_{n2}A_{n3}\dots A_{nn} \end{bmatrix} \begin{bmatrix} p1 \\ p2 \\ \dots \\ pn \end{bmatrix} + (1-d) \begin{pmatrix} 1 \\ 1 \\ \dots \\ 1 \end{pmatrix}$$

Where A is the transition matrix of the web graph.

$$A_{ij} = \begin{cases} \frac{1}{O_i} & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

So the final formula of PR is:

$PR\,(p_i) = (1-d) + d \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)}$

To have normalized value of PageRank the factor 1-d is divided by the number of documents (N) in the collection.
Our goal is to run the algorithm on these pages to obtained their ranks. Run the algorithm on the web space and using an approach based on recursively update for hundred thousand pages will be so expensive and time consuming. MapReduce approach will tackle the problem by taking the advantage of running on a cluster (parallelization) and scaled up to very large link-graphs (large number of pages). This approach is based on this following structure:

- *"Map: For each node i, calculate vote (Ri/Di) for each out-link of i and propagate to adjacent nodes.*
- *Reduce: For each node i, sum the upcoming votes and update Rank value (Ri).*
- *Repeat this Map-Reduce step until Rank values converge (stable or within a margin)"* [5]

3 Centrality measures

Centrality measures are a powerful tool to analyse the network, retrieve some node's properties and extract some characteristics that make a node interesting or particular important. The most used centrality measures are:

- **Betweennes centrality**;
- **Degree centrality**;
- **Closeness centrality**.

With the following sections will be introduced more in detail every measure to explain more in detail the meaning and the utility for the network analysis.

3.1 Betweennes centrality

"The betweenness centrality captures how much a given node (hereby denoted u) is in-between others. This metric is measured with the number of shortest paths (between any couple of nodes in the graphs) that passes through the target node. This score is moderated by the total number of shortest paths existing between any couple of nodes of the graph (the target node would have a high betweenness centrality if it appears in many shortest paths)" [2].
The following formula explain how to calculate the Betweennes centrality measure:
$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}$
where σ_{st} is the total number of shortest paths from node s to node t and $\sigma_{st}(v)$ is the number of those paths that pass through v.

3.2 Degree centrality

"Degree centrality is one of the easiest to calculate. The degree centrality of a node is simply its degree—the number of edges it has. The higher the degree, the more central the node is" [4]. We are going to use Freeman's general formula to compute degree centralization. First, find the vertex with the highest degree (We will call it v*), then define $H = (\|V\| - 1)(\|V\| - 2)$.
The following formula explain how to calculate the Degree centrality measure:
$C_D(G) = \sum_{v \in G} \frac{|\deg(v_*) - \deg(v)|}{|H|}$

3.3 Closeness centrality

"Closeness centrality indicates how close a node is to all other nodes in the network. It is calculated as the average of the shortest path length from the node to every other node in the network" [3]. Closeness centrality can be viewed as the efficiency of a vertex in spreading information to all other vertices
The following formula explain how to calculate the Closeness centrality measure:
$C(v) = \sum_{w \in G} \frac{1}{d(v,w)}$
In other words, if the sum of the distances is large, then the closeness is small and vice versa. A vertex with a high closeness centrality would mean it has close relationships with many vertices.

4 Implementation's detail

4.1 Environment, Framework & other software

The software that was used to write, compile, run and debug codes is **JupyterLab**. JupyterLab is a web-based interactive development environment for Jupyter notebooks, code, and data.

In order to implement the PageRank algorithm were used a set of libraries and framework likes:

- **ApacheSpark:** *"Open-source cluster-computing framework, built around speed, ease of use, and streaming analytics whereas Python is a general-purpose, high-level programming language"* [1]
- **GraphFrames:** *"GraphFrames is a package for Apache Spark which provides DataFrame-based Graphs. It provides high-level APIs in Scala, Java, and Python."* [9]
- **NetworkX:** *"NetworkX is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks."*[10]

It is also used Cytoscape that is an open source software platform for visualizing, analyze complex networks and see important and interesting nodes's characteristics and integrating these with any type of attribute data. We decided to use Cytoscape

to retrieve some network's properties analyzing for example how the network is structured or how many links the nodes have. In the next sections will be reported the analysis obtained using this software.

4.2 Algorithms

The PageRank algorithm is based on the use of the pyspark library. The dataset for this project is a webgraph consisting of 875713 nodes and 5105039 edges. In the file used file the first 4 lines are the dataset's descritpion. The algorithm is created taking inspiration from the code that you can find in this link. At first it's necessary build the right configuration for running pySpark, the following two rows set the basic configurations for running a PySpark application.

```
conf = SparkConf().setAppName("app").setMaster("local")
sc = SparkContext(conf=conf)
```

For each row the first value corresponds to the source node ID followed by a separation character (Tab) and a value representing the recipient node ID The algorithm can be summarized with the following points:

1. Reading the file that contained all the nodes and edges.
2. Skip the first four lines of the files.
3. Reading all the links, through the map function an RDD structure is made in which every node is associated with the list of the nodes connected to it.
4. Assigning rank equal to 1 for all nodes.
5. Calculation of new contributions, for each node the list of associated nodes are taken, its current PageRank value is taken and divided by the number of nodes present in the list (number of outgoing edges), this value is assigned as contributes to the received nodes. This step is carried out through a Map function that calls, for the calculation of the contribution, the function **computeContribs**.
6. It has calculated the PageRank of each node as the sum of all its contributions through pyspark function called **reduceByKey(add)** multiplied by d(damping values) added to the value of alpha (1-d).

7. It is calculated the variable that determines the convergence of the algorithm (delta) effected for each iteration the difference between the total sum of the PageRank values of the previous iteration with the total sum of the PageRank values of the current iteration.

8. The points 5, 6, 7 are repeated until the delta value is less than $1 * 10^{-6}$.

9. PageRank values are sorted in descending order and saved to a CSV file.

To avoid the problem of missing nodes, who don't have any incoming links, that the algorithm just presented had, we used a support structure (called temp_struct) with all nodes that have at least one outgoing link. In each iteration, using this structure and some operations we added the lost nodes setting their contribution to 0. This way at the end of all iterations they will have a value equal to 1-d.

5 Analysis and evaluation

The PageRank algorithm was performed with several alpha values. The alpha value change, the scores and the ranks of web pages, a lower damping factors decrease the likelihood of following long paths of links without taking a random jump and thus increase the contribution to a node's score and rank of its more immediate ancestors.[7]
As you can see in table 1,*"the number of iterations taken by PageRank method grows on increasing the value of d and required more numerical precision to converge."* [**Dampingvalue**] The PageRank formula has been used without normalization,

alpha_value	n_iteration
0.10	222
0.15	149
0.50	35

Table 1: Number of iteration for different value of alpha

except for alpha equal to 0.15 that the PageRank has been calculated also with the formula with output score values between 0 and 1 (normalization does not affect rank in any way but only on the score scale)
To better understand the structure of the graph, several aspects have been analyzed.

The next analysis, that will be presented, will be carried out through the Graphframes and Networkx packages.

5.1 Indegree anlysis, connected components

The first analysis concerns the node's degree distribution within the network. In particular, the analysis focused on the degree of incoming links as more significant for the comparison with PageRank (the latter is influenced only by incoming links). From our analysis it is clear that the node's degree distribution is concentrated in the range 0-790. In particular there are 714380 nodes with inDegree in the band 0-790 so more than **95%** of the nodes of the networks.

There are few nodes with high degree in particular there is a node with inDegree equal to 6320, what you might expect is that such node has the highest PageRank, however if you go to verify the value of PageRank (with d equal to 0.85) this node has one of the higher values of PageRank but not the highest, this means that it has many incoming links but coming from nodes with low PageRank values, so from pages with no particular relevance. The node that has the second highest value of PR (id 41909) has as inDegree equal to 4129 so, even if it has 2000 less incoming links respect the node with the highest inDegree it has an higher PR so this means that the incoming links are comning from more important nodes.

Analyzing better the 0-790 band in the table 2 we can see as there are 401437 nodes with inDegree equal to 1 or 2, and even if these nodes have only 1 or 2 incoming links this doesn't mean that the PageRank's values will be low, this because the 1 or 2 incoming links could be coming from high importance nodes.

Proceeding in the analysis, we calculated the connected components through the graphFrames's function **connectedComponents**. The given analysis revealed that there is a connected component of 647500 nodes, so this means that there is a giantComponent that represents 87% of nodes of entire network. So if you start from a page represented by a node present in this giantComponent you would be able to reach most of the nodes(pages) of the network. The remaining nodes are isolated in small connected components.

ne_bin	inDegree
(0, 2]	401437
(2, 131]	310703
(131, 263]	1593
(263, 394]	358
(394, 526]	161
(526, 657]	78
(657, 790]	50

Table 2: In degree distribution in band in degree(0-790)

5.2 Community detection and analysis

Proceeding with the analysis we calculated the different centrality's measures and because they are computationally complex we decided to proceed with the network's clustering operation and then analyze one of the most significant community. To clustering the network was used graphFrames's labelPropagation, *" function where each node in the network is initially assigned to its own community. At every superstep, nodes send their community affiliation to all neighbors and update their state to the mode community affiliation of incoming messages"*[9]
In this way we calculated the various communities. For the next analysis we decided to focus only on one community to make the results more readable and easy to interpret.
The new graph that we analyzed consists of all the nodes belonging to a particular community and all the nodes to which these nodes pointed even if the latter belong to another community. The particular community that we have chosen is the community that belongs the node with id equal to 41909 that we have already talked about.

For this new graph we have calculated several measurements of centrality: betweennesss, degree, closeness (presented in the section) through the netowrkX library The software **Cytoscape** was used for the interpretation of the results and the comparison, this software also helps with the analysis of the network.

Some analyses are given:

- Nodes of network: 4243
- Edges of network: 9964

- Network diameter: 5
- Network density: 0.001

Density refers to the "connections" between nodes. Density is defined as the number of connections a node has, divided by the total possible connections a node could have. The value of density for this portion of graph is low in fact as previously anticipated many nodes have low grade(1 or 2) so as expected the network is not strongly connected, quite common situation for a graph that represents a web network. The diameter of a graph is the largest distance between pairs of nodes of the graph so in the portion of networks analyzed with a average of maximum 5 steps starting from a nodes you could get to a node in the opposite side of network.

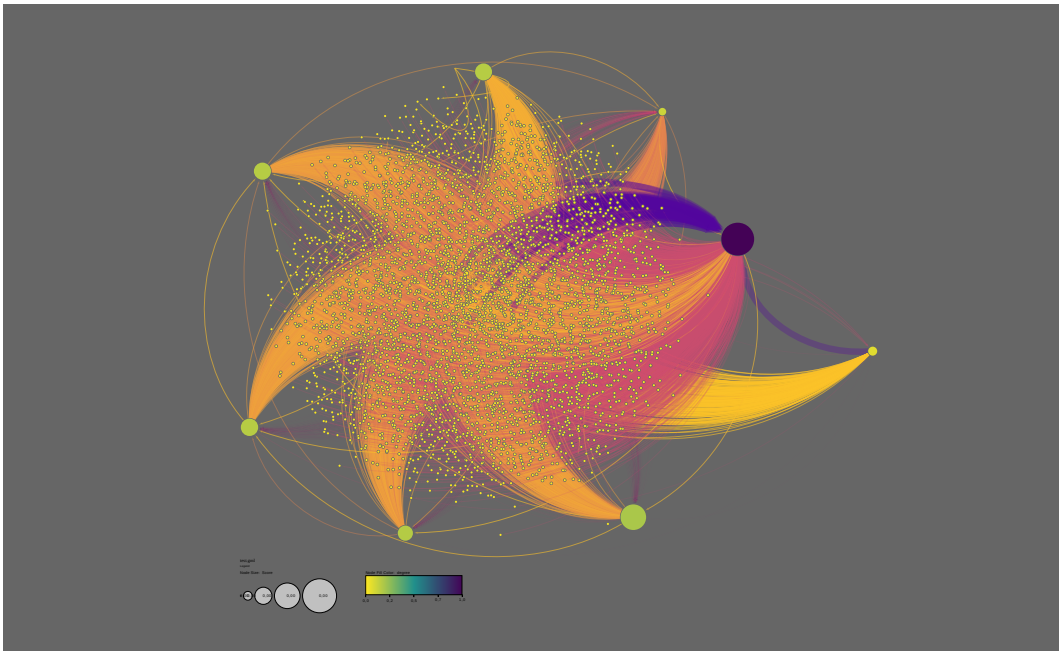

Figure 1: Cummonity's graph

In the image 1 is represented the sub-graph in particular

- Nodes' size depends on PageRank value, higher is the value of the PageRank, bigger is the node;
- Nodes' color depends on degree value, higher is the value of the degree, darker is the color;
- Edges' color depends on betweenness value, higher is the value of the betweenness, darker is the color.

As you can see from the figure just reported 1 are labeled with numbers those nodes that have a higher PageRank value than all the others. The association between the number and the node in no way reflects the ranking in which the nodes are ordered relative to the value of PageRank but, have been labeled only to provide a clearer analysis later. The node with the highest PageRank value is node number 1 definitely influenced by the large amount of incoming edges and the incoming edger coming from node 2 which has a decent PageRank value and for this reason a positive influence on node 1. The PageRank depends not only on the number of incoming edges but also on the quality of them, there are 7 other nodes with high PageRank value and these nodes are connected to each other and it is for this reason that they have high PageRank value, eg: Node 3 has an incoming border from node 1 this border positively influences its Pagerank value as it comes from a node with a high value. The only node among the 8 considered that does not have any incoming edge deriving from one of the remaining 7 is node 2 but, it has an inDegree value quite high that allows it to have a high PageRank value. Analyzing the betweennes of the edges we notice how the edges entering in node 1 have all high betweennes value (dark edges) this is because it is a central node of the network and many paths through the nodes will pass from node1 so, a lot information will pass from that node. Also from node 2 to node 1, there is an arc with betweenness very high because node 2 is the predecessor of node 1 which is central to the betweenness with a value of **0.9714** which turns out to be the highest value of all. As you can see, however, the other bows with a high value of betweenness come from bows with a low value of Pagerank (small nodes) in fact the betweenness, unlike the Pagerank, does not evaluate the quality but only the quantity of paths passed by that arc. If you analyze the authorities values of these 8 nodes they have high values because

are pages containing useful information. Also for closeness centrality measurement the nodes with highest betweenness and degree centrality are the nodes with higher closeness value, this happened because closeness measures the centrality of the nodes in the network and it answer the question :"How fast can,from this node reach every node in the network?" so for a node with a lot of links is easier reach other nodes. Both measures (Closeness and betweenness) are strong correlated because are based on the concept of the shortest path.

6 Conclusion

In this work we have explored the operation of the PageRank algorithm in a real scenario, deepening the use of the damping factor and when this affects. By representing web pages as a graph we were able to analyze the structure and properties. The main characteristics recovered are how much the network (analyzed) turns out to be composed of a giant component and therefore little nodes, pages, turn out to be disconnected from the main member. The graph has a low density in how much being a real network a page will point to a limited number of pages regarding the totality of the present pages in the network. Analyzing in detail a sub-portion were calculated the classic measurements of centrality, realizing how much these give importance only to the quantity of the links to determine the centrality of the nodes while the Pagerank turns out sure to be a more complete measure for a web graph seen that it holds account also of the quality of the links.

References

[1] Swatee Chand. "PySpark Programming – Integrating Speed With Simplicity". In: *https://www.edureka.co/blog/pyspark-programming/: :text= PySpark%20is%20the%20collaboration%20of,%2C%20high-level%20programming%20language* (2020).

[2] Rony Germon Charles Perez. "Betweenness Centrality". In: *https://www.sciencedirect.com/topics/computer-science/betweenness-centrality* (2016).

[3] Jennifer Golbeck. "Closeness Centrality". In: *https://www.sciencedirect.com/topics/computer-science/closeness-centrality* (2013).

[4] Jennifer Golbeck. “Degree Centrality”. In: *https://www.sciencedirect.com/topics/computer-science/degree-centrality* (2013).

[5] Taylan Kabbani. “PageRank on MapReduce”. In: *https://medium.com/swlh/pagerank-on-mapreduce-55bcb76d1c99* (2020).

[6] Chandler long. “google’s infamous pageranking algorithm”. In: *http://www.chandlerlong.com/pagerankBlog.html* (2018).

[7] Enoch Peserico Marco Bressan. “Choose the damping, choose the ranking?” In: *https://www.sciencedirect.com/science/article/pii/S1570866709000926* (2010).

[8] Margaret Rouse. “PageRank”. In: *https://whatis.techtarget.com/definition/PageRank* (2017).

[9] Unknown. “GraphFrames User Guide”. In: *https://graphframes.github.io/graphframes/docs/$_site$/user − guide.htmlgraphframes − user − guide* (2020).

[10] Unknown. “NetworkX”. In: *https://networkx.org* (2020).