Module 6

Page Rank Algorithm

PageRank is an algorithm developed by Google founders Larry Page and Sergey Brin that measures the relevance or importance of web pages on the Internet. The PageRank algorithm treats the web as a vast network of interconnected pages. Each page is represented on the web as a node with links between pages at the edges. The basic principle of PageRank is that a page is considered more important if other vital pages link it. The algorithm determines the initial PageRank value for each web page. This initial value can be uniform or based on certain factors, such as the number of incoming links to the page. The algorithm then repeatedly calculates the PageRank value of each page, taking into account the PageRank value of the pages that are related to the pages. During each iteration, the PageRank value of the page is updated based on the sum of the PageRank values of the incoming links. Pages with more inbound links have a more significant impact on the landing page's PageRank.

Page Rank Algorithm

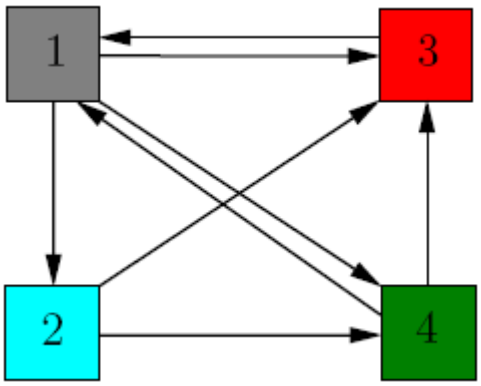
The PageRank algorithm assigns a numerical **value called a** PageRank **score** to each web page in a linked page network. **Points indicate** the **relevance or** importance of the page **online. The** algorithm works step by **step:**

1. **Initialization:** The algorithm begins by determiningthe initial PageRank value of each web page. Typically, this initial value is set uniformly across all pages so that every page has the same initial value.
2. **Link analysis:**The algorithm analyzes the links between web pages. It considers both inbound links (links pointing to a page) and outbound links (links from a page to other pages). Pages with more inbound links are considered more important because they are believed to receive recommendations or votes of trust from other important pages.
3. **Iterative calculation:**The algorithm repeatedly updates the PageRank score of each page based on the PageRank score of the related pages. During each iteration, the PageRank of a page is recalculated, taking into account the PageRank contribution of its incoming links. Damping factor: a damping factor (typically 0.85) is introduced to avoid infinite loops and ensure the algorithm This indicates that the user will likely continue browsing by following a link on the current page rather than jumping to a random page. The damping factor helps to evenly distribute the importance and block the entire PageRank value on a single page.
4. **Rank Distribution:**As the algorithm progresses, the PageRank of the page is distributed among the outgoing links. For example, if a page has a high PageRank and many outbound links, each link will contribute to the overall impact of the page. This division ensures that the importance of linked pages is shared.
5. **Convergence:**The iterative process continues until the PageRank score stabilizes or converges. Convergence occurs when the difference in PageRank scores between successive iterations falls below a certain threshold. At this point, the algorithm has reached a stable ranking, and the PageRank scores indicate the relative importance of each web page.
6. **Ranking and Display:**Pages are ranked based on their final PageRank scores. Pages with a higher PageRank score are considered more influential or essential Search engines can use these points to display search results, so pages with higher rankings are usually shown closer to the top. By considering the link structure and updating the PageRank score iteratively, the algorithm

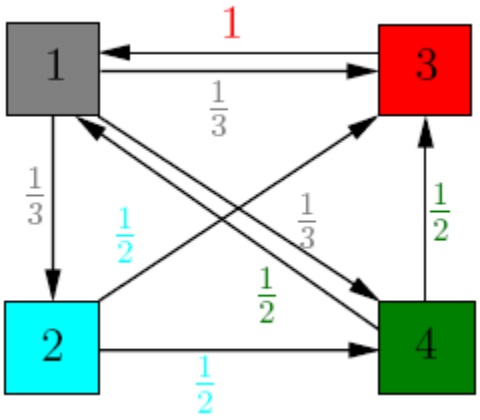
effectively measures the importance of web pages relative to others. It allows ranking pages based on their popularity and influence, helping to develop more accurate and relevant search engines.

For Example,

Consider below graph and we have to calculate page Rank



Node 1 has 3 outgoing edges, so it will pass on $\frac{1}{3}$ of its importance to each of the other 3 nodes. Node 3 has only one outgoing edge, so it will pass on all of its importance to node 1. In general, if a node has *k* outgoing edges, it will pass on $\frac{1}{k}$ of its importance to each of the nodes that it links to. Let us better visualize the process by assigning weights to each edge.



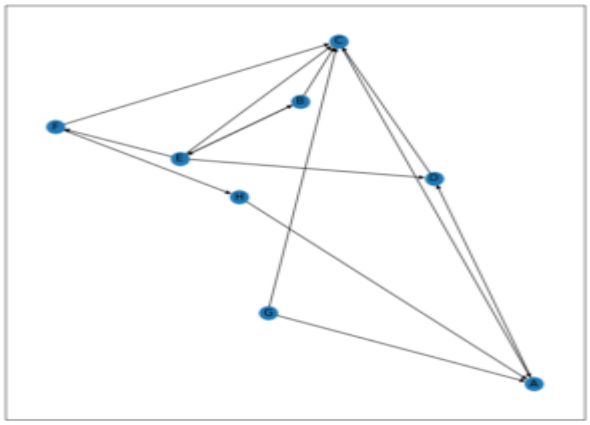
Let us denote by *A* the transition matrix of the graph, $A = \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix}$.

Hyperlink Induced Topic Search (HITS)

Hyperlink Induced Topic Search (HITS) Algorithm is a Link Analysis Algorithm that rates webpages, developed by Jon Kleinberg. This algorithm is used to the web link-structures to discover and rank the webpages relevant for a particular search.
HITS uses hubs and authorities to define a recursive relationship between webpages. Before understanding the HITS Algorithm, we first need to know about Hubs and Authorities.

- Given a query to a Search Engine, the set of highly relevant web pages are called **Roots**. They are potential **Authorities**.
- Pages that are not very relevant but point to pages in the Root are called **Hubs**. Thus, an Authority is a page that many hubs link to whereas a Hub is a page that links to many authorities.

For Example, Let us consider the following Graph:



On running HITS Algorithm with k = 3 (without Normalization),

Initially,

Hub Scores:	Authority Scores:
A -> 1	A -> 1
B -> 1	B -> 1
C -> 1	C -> 1
D -> 1	D -> 1
E -> 1	E -> 1
F -> 1	F -> 1
G -> 1	G -> 1
H -> 1	H -> 1

After 1st iteration,

Hub Scores:	Authority Scores:
A -> 1	A -> 3
B -> 2	B -> 2
C -> 1	C -> 4
D -> 2	D -> 2
E -> 4	E -> 1
F -> 1	F -> 1
G -> 2	G -> 0

H -> 1 H -> 1

After 2nd iteration,
Hub Scores: Authority Scores:
A -> 2 A -> 4
B -> 5 B -> 6
C -> 3 C -> 7
D -> 6 D -> 5
E -> 9 E -> 2
F -> 1 F -> 4
G -> 7 G -> 0
H -> 3 H -> 1

After 3rd iteration,
Hub Scores: Authority Scores:
A -> 5 A -> 13
B -> 9 B -> 15
C -> 4 C -> 27
D -> 13 D -> 11
E -> 22 E -> 5
F -> 1 F -> 9
G -> 11 G -> 0
H -> 4 H -> 3