

PageRank Algorithm

The PageRank algorithm is designed to operate on a directed graph, where the nodes represent web pages and the edges represent links between pages. It is designed to assign a rank to each individual web page, where this rank is an indication of importance/authority. Pages with higher ranks are considered to be more important. Therefore, search engines that use the PageRank algorithm typically yield the pages of highest rank at the top of the results list.

Each node's page rank is a function of its inbound neighbors' page ranks. Effectively, if we have n nodes in our graph, this gives us a system of n linear equations in n unknowns. This system could be solved directly, but is extremely space inefficient, as it would require storing a (likely sparse) $n \times n$ matrix. Instead, the PageRank algorithm computes ranks using an iterative approach that starts by assigning the same importance to all pages and gradually adjusts these values until it approaches the correct solution.

The PageRank algorithm takes two inputs: a graph, g , and a damping factor, d . We will use $PageRank_k(p_i)$ to denote the rank of a given page p_i in the k^{th} iteration. The PageRank algorithm proceeds as follows:

1. First, we need to remove all sink nodes from the graph by adding outbound edges from those nodes to every other node in the graph.
2. Next, we initialize every page's rank as follows: $PageRank_0(p_i) = 1/n$.
3. During each subsequent iteration, we will update the pages' ranks "simultaneously" using a snapshot of their ranks from the previous iteration. Specifically, in each iteration, each page's rank is updated as follows:

$$PageRank_k(p_i) = \frac{1-d}{n} + (d \times (\sum_{p_j \rightarrow p_i} (PageRank_{k-1}(p_j) / outdegree(p_j)))).$$

4. After each iteration, we will compute $delta_k = \sum_{p_i \in g} |PageRank_k(p_i) - PageRank_{k-1}(p_i)|$. If $delta_k$ is less than 10^{-8} , we will declare that the algorithm has converged and return the results in the form of a dictionary mapping each page (node) to its PageRank from the final iteration.

