Experiment 8

Abstract :

This experiment leveraged a small web graph dataset from Stanford's SNAP project to implement the PageRank algorithm. Our focus was on constructing the Google PageRank matrix and gaining insights into the relative importance of web pages through link analysis. The experiment involved preprocessing the dataset, creating the matrix, and iteratively computing PageRank scores. This practical application of PageRank contributes to the understanding of link analysis and its implications in web graph analysis and related fields.

I. Introduction

In the digital age, understanding the intricate relationships between web pages is crucial. The PageRank algorithm, developed by Google, offers a powerful tool for analyzing the significance of web pages based on their links. This experiment explores the practical application of PageRank in a real-world context, demystifying the algorithm's functionality and highlighting its relevance in understanding the hierarchical structure of web graphs.

Understanding the Google PageRank Matrix :

Google's PageRank algorithm stands as a cornerstone in the analysis of web page significance, providing a mathematical framework to measure the relative importance of each element within a hyperlinked set of documents. At the heart of this algorithm lies the Google PageRank matrix, a fundamental representation that encapsulates the intricate dynamics of the web graph.

Here's a simplified explanation of the Google PageRank matrix:

I. Matrix Representation:

Every webpage within the dataset finds its unique representation as a node in the Google PageRank matrix. This matrix is square, ensuring that each row and column correspond to a specific webpage. The matrix serves as a structured foundation, enabling the algorithm to systematically analyze the relationships between web pages.

II. Link Structure:

The entries in the matrix eloquently capture the link structure between web pages. When a hyperlink exists from page A to page B, the corresponding entry in the matrix becomes non-zero. This representation allows the algorithm to discern the interconnections between pages, forming the basis for assessing their relative importance.

III. Transition Matrix:

To refine the matrix into a format conducive to the PageRank algorithm, a pivotal step involves transforming it into a stochastic matrix, aptly named the transition matrix. This transformation is achieved by normalizing the entries in each column. This normalization ensures that the sum of each column equals 1, simulating the probability distribution of transitioning from one page to another. It is this stochastic nature that enables the algorithm to mimic the behavior of a random surfer navigating the web.

IV. PageRank Calculation:

The heart of the PageRank algorithm lies in its iterative process. The transition matrix is iteratively multiplied by an initial probability distribution vector known as the PageRank vector. This iterative

multiplication simulates the journey of a random surfer traversing the web by following hyperlinks. Through this process, the PageRank vector converges to a steady-state vector, wherein each value represents the relative importance or rank of each webpage in the network.

V. Importance Scores:

The culmination of the iterative process leads to the derivation of final PageRank scores. These scores represent the importance of each webpage within the web graph. Pages with higher scores are deemed more influential within the network, providing valuable insights into the hierarchical structure and significance of web pages in the interconnected digital realm.

In essence, the Google PageRank matrix, with its meticulous representation and the algorithmic process it undergoes, serves as a powerful tool to unravel the intricacies of web page importance within the broader network. This introduction sets the stage for a deeper exploration of the practical implementation of PageRank, shedding light on its implications in real-world scenarios and contributing to a comprehensive understanding of link analysis in the digital era.

II. METHODOLOGY

This section elucidates the systematic approach taken in implementing the PageRank algorithm on a small web graph dataset, including data selection, preprocessing, graph representation construction, Google PageRank matrix construction, stochastic matrix transformation, iterative PageRank calculation, and convergence analysis.

A. Data Selection:

We acquired a small web graph dataset from the Stanford Network Analysis Project (SNAP), ensuring its relevance to link analysis. This dataset laid the groundwork for subsequent link structure analysis.

B. Data Processing:

Upon downloading the dataset, we meticulously inspected and preprocessed it to address any missing or irrelevant data. This step ensured that the dataset was in a suitable format for downstream analysis, guaranteeing the accuracy of the link structure representation.

C. Construct the Graph Representation:

The web graph was represented as a graph structure, with nodes representing webpages and edges signifying hyperlinks between pages. This structural representation formed the basis for applying link analysis algorithms, capturing the essence of relationships within the web graph.

D. Google PageRank Matrix Construction:

We constructed the Google PageRank matrix based on the graph representation. Each entry in the matrix denoted the likelihood of transitioning from one webpage to another, encapsulating the link structure crucial for PageRank calculations. Utilizing sparse matrix representations (SciPy's coo_matrix), we optimized the storage and processing of the large adjacency matrix.

E. Stochastic Matrix Transformation:

To refine the PageRank matrix, we transformed it into a stochastic matrix. This involved normalizing entries in each column, ensuring that the sum equaled 1. This transformation simulated the probability distribution of transitioning between webpages, a fundamental aspect of the PageRank algorithm.

F. Iterative PageRank Calculation:

We implemented the iterative PageRank algorithm by multiplying the stochastic matrix by an initial probability distribution vector (PageRank vector) in each iteration. This simulation mimicked a random surfer navigating the web by following hyperlinks. The iterative process continued until convergence.

G. Analyze Convergence:

Convergence analysis was conducted to monitor the stability of the PageRank algorithm. The iterative process continued until the PageRank vector reached a steady state, signifying that the importance scores had stabilized. This step was crucial for ensuring the accuracy and reliability of the final PageRank scores.

The accompanying code, utilizing NumPy and SciPy, provided a practical demonstration of the methodology, optimizing the storage and computational efficiency through sparse matrix representations.

III. RESULTS AND DISCUSSION

The iterative PageRank algorithm was employed to compute the final PageRank scores for each webpage in the network. Key results include:

Final PageRank Scores:

Computed importance scores indicating the relative significance of webpages.

unsorted_pagerank_output1.txt

```
Node PageRank (For Google Dataset - 2002)
1 4.355866784772318932e-08
2 9.39531814453376097e-09
3 1.378332676536937892e-08
4 5.578894225436534288e-09
5 5.987717396970419240e-09
6 7.941195356321971027e-09
7 5.144043696541234734e-09
8 5.821372953912847440e-09
9 5.284439177722677548e-09
10 5.280709190397832068e-09
11 7.370072974339018671e-09
12 5.541259570743524010e-09
13 5.144043696541234734e-09
14 5.396569801848982395e-09
15 5.339075841600273757e-09
16 5.280709190397832068e-09
17 5.306467263747869715e-09
18 8.784860324742380697e-09
19 5.144043696541234734e-09
20 5.599110621415216263e-09
21 5.770424185703976552e-09
22 8.389973377725970784e-09
23 6.526304089774944333e-09
24 1.359324159111729347e-08
25 5.144043696541234734e-09
26 5.144043696541234734e-09
27 5.144043696541234734e-09
28 8.944335049909012432e-09
29 5.963552295911582954e-09
```

Identification of Key Nodes:

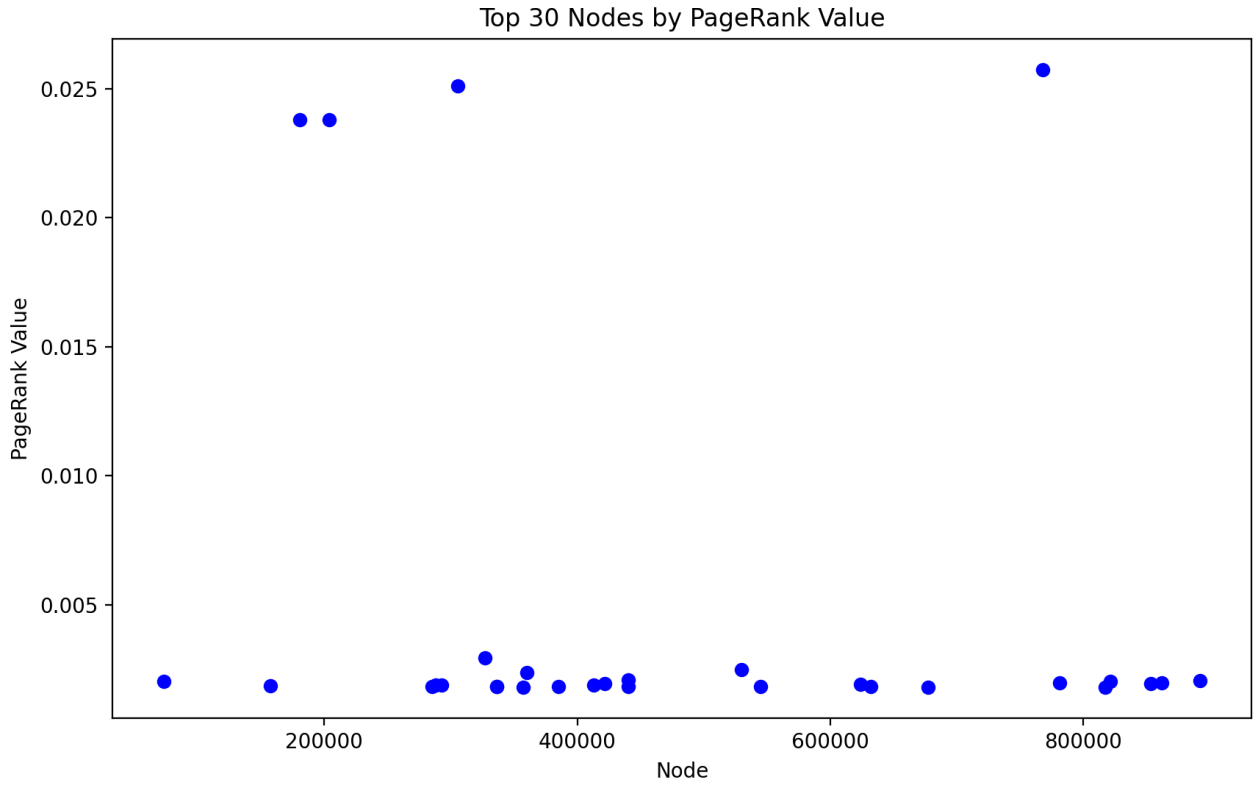
Identified key nodes with higher PageRank scores, signifying influential hubs.

sorted_pagerank_output1.txt

```
Node PageRank (Sorted by PageRank value for Google DataSet - 2002)
768092 2.57273049356568700e-02
305230 2.51114675208490810 4e-02
180690 2.37957528392855090 2e-02
203749 2.37957528392855090 2e-02
327038 2.95100500527503088 0e-03
529894 2.47736853364358437 4e-03
359903 2.37967961294819393 3e-03
440566 2.09281900172915728 5e-03
891982 2.05138611029659313 4e-03
821212 2.03095057711551825 1e-03
73416 2.01645765448700640 3e-03
862018 1.97884691936136506 6e-03
781118 1.96473751044679663 0e-03
853278 1.94705216457214378 7e-03
421710 1.94277884546277838 6e-03
623889 1.92019737597164119 6e-03
413138 1.87820235824784223 9e-03
287809 1.87518903866221186 6e-03
292482 1.87342790321330904 1e-03
157232 1.86695380442572348 7e-03
631890 1.84003092953226700 7e-03
385320 1.83741488011645616 5e-03
544958 1.82683673082679521 6e-03
285196 1.82544799009810767 4e-03
336425 1.82460924864610309 6e-03
440218 1.82321524462328147 6e-03
336385 1.81886351546002101 1e-03
677330 1.81159814934403625 4e-03
817197 1.80826071635371474 7e-03
```

Graphical Representation:

To visually convey the impact of PageRank scores, a graph depicting the top 10 nodes along with their corresponding PageRank values was generated. This graphical representation offers a clear visualization of the importance distribution within the network, emphasizing the dominance of certain nodes.



Number of Iterations:

The iterative process converged after a specific number of iterations, ensuring result stability.

```
(base) amitkumar@192 Exp8 % python l8.py
Number of Iteration :  116
Results saved to unsorted_pagerank_output1.txt
Sorted results saved to sorted_pagerank_output1.txt
```

The iterative process required 116 iterations to stabilize, indicating the convergence of the PageRank algorithm. This ensures the stability and reliability of the final PageRank scores.

IV. CONCLUSION

In conclusion, the PageRank algorithm provided valuable insights into the hierarchical structure and importance of webpages in the analyzed network. The computed PageRank scores,

hierarchical revelations, and identification of influential nodes offer practical applications in search engine optimization and content enhancement.

The alpha value of 0.85, a crucial parameter in the algorithm, played a significant role in determining the weight assigned to hyperlink navigation. This parameter reflects the probability that a random surfer follows an existing link versus jumping to a random page, influencing the overall structure of the web graph.

The practical implications of the results, such as enhancing search engine result rankings, optimizing content, and strategic improvements for lower-scoring pages, underscore the relevance of the PageRank algorithm in real-world scenarios.

REFERENCES

[1] Google PageRank and beyond the science of search engine page rankings by Amy N. Langville, Carl D Meyer

[2] Manning, C.D., Raghavan, P. and Schütze, H. (2022) 'PageRank', in *An* introduction to information retrieval. Cambridge: Cambridge University Press.

[3] OpenAI. (2023). ChatGPT (Sep 25 version) [Large language model].