WEEK-10

Implementation of PageRank on Scholarly Citation Network.

AIM:

To implement Page Rank on scholarity Citation Network.

DESCRIPTION:

What is PageRank?

- PageRank is an algorithm originally developed by Google founders Larry Page and Sergey Brin.
- It measures the *importance* of nodes in a directed graph, based on the idea that a node (like a web page or research paper) is important if other important nodes link (or cite) it.
- In a scholarly citation network:
    - **Nodes = Research papers**
    - **Edges = Citations (directed links from citing paper to cited paper) Use of**

PageRank in citation networks

- To identify **influential research papers**.
- A paper is important not just because it has many citations, but because it is cited by other important papers.
- It helps rank scholarly articles in terms of *impact* rather than just *counting citations.*

Example

Suppose we have a small citation network:

- Paper1 → cites Paper2, Paper3
- Paper2 → cites Paper3
- Paper3 → cites Paper1

This forms a cycle. Running PageRank will eventually distribute scores showing which paper is most central in the citation loop.

PROGRAM:

```
pip install --upgrade numpy scipy network

import networkx as nx

# Example scholarly citation network

# Each node is a paper, edges represent citations

citations = {

    "Paper1": ["Paper2", "Paper3"],

    "Paper2": ["Paper3"],

    "Paper3": ["Paper1"],
    "Paper4": ["Paper2", "Paper3"],

    "Paper5": ["Paper3", "Paper4"]

}

# Build directed graph G = nx.DiGraph()

for paper, cited_papers in citations.items():

    for cited in cited_papers:

        G.add_edge(paper, cited)

# Compute PageRank manually (no scipy backend needed) pagerank_scores = nx.pagerank(G,

alpha=0.85, max_iter=100) print("\n PageRank Scores:")

for paper, score in pagerank_scores.items():

    print(f"{paper}: {score:.4f}")
```

OUTPUT:

```
PageRank Scores:
Paper1: 0.3515
Paper2: 0.1975
Paper3: 0.3782
Paper4: 0.0428
Paper5: 0.0300
```