MATHEMATICS FOR COMPUTING & ELEMENTS OF COMPUTING

Page rank algorithm and Google Search Engine

Group - 13

Team Members

Adhwaidh K	-	CB.SC.U4AIE24003
Adith S	-	CB.SC.U4AIE24004
Chaitanya Varma	-	CB.SC.U4AIE24017

Introduction

- The PageRank algorithm, is used to rank web pages based on their importance in a network (web graph).
- It simulates the behavior of a "random surfer" who randomly clicks links on the web.
- This project implements and visualizes PageRank using multiple techniques:
  - Algebraic method (Gauss elimination and matrix inversion)
  - Power iteration method
  - NetworkX library
  - Web crawling and graph creation using BeautifulSoup

Objectives

- To understand and compare multiple approaches to compute PageRank.
- To visualize PageRank values for various networks and web structures.
- To demonstrate real-time PageRank computation by crawling web pages.
- To track convergence and rank stability over iterations.
- To export and analyze ranked pages for external use.

Literature Review

S. No.	Title	Advantages	Limitations
1.	The Anatomy of a Large-Scale Hypertextual Web Search Engine	- Introduced a scalable method for ranking web pages. - High accuracy for determining page importance.	- Computationally expensive for large datasets
2.	Google's PageRank and Beyond	- Provided mathematical rigor for PageRank and iterative methods - Focus on matrix algebra optimizations	- Heavy computational burden for extremely large matrices

Literature Review

S. No.	Title	Advantages	Limitations
3.	Network Analysis in Python	- Easy implementation of PageRank with built-in functions - Visualization features for better interpretation	- Limited customization for complex variations of PageRank - Performance issues with very large graphs

Research Gaps

- Many PageRank implementations do not focus on multi-method comparisons or visual analysis.
- Few projects demonstrate real-time crawling and PageRank computation with link-based graph visualization.
- There's limited understanding of convergence behavior and how quickly PageRank stabilizes across methods.
- Exporting and utilizing ranked data for further decision-making is underexplored.

Problem Statement

- Too many web pages make it hard to identify which ones are most important or relevant.
- Existing systems don’t fully use link structures between pages to rank them effectively.
- Lack of efficient algorithms to rank pages based on popularity or importance.
- Difficulty in implementing and visualizing PageRank on real-time or live web data.
- No clear comparison between different methods of calculating PageRank (like Gauss Elimination, Matrix Multiplication, Power Iteration, etc.).

Methodology

Algebraic Method:

- Constructed transition matrix A based on a fixed web graph.
- Used **G**auss elimination and inverse matrix multiplication to compute steady-state ranks.

Power Iteration Method:

- Initialized a random probability vector.
- Applied iterative multiplication with the adjusted transition matrix to simulate surfer behavior.
- Tracked rank convergence over 100 iterations.

NetworkX Implementation:

- Built a custom directed graph of 10 nodes.
- Applied nx.pagerank() with damping factor ($\alpha = 0.85$).
- Visualized PageRanks using pie charts and circular layouts.

Web Crawler Integration:

- Used requests BeautifulSoup to crawl real web pages starting from a Wikipedia article.
- Constructed a directed graph from crawled links.
- Implemented a custom PageRank algorithm.
- Tracked convergence history and exported results.

Future Scope

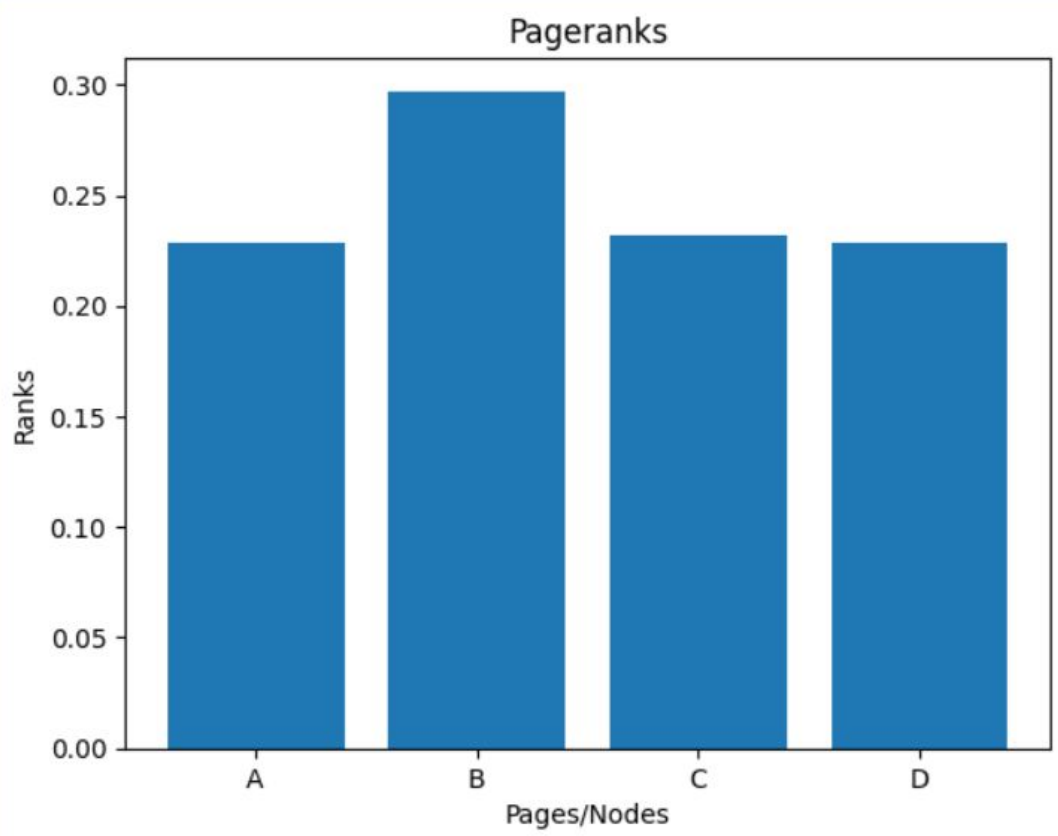
- Extend to larger web crawls (100+ pages) with optimizations (e.g., parallel crawling).
- Apply Topic-Sensitive PageRank or Personalized PageRank for user-centered search.
- Integrate textual relevance and content analysis alongside link structure.
- Deploy the project as a web-based dashboard for dynamic visualization.
- Analyze link spam detection using anomalies in PageRank values.
- Compare results with Google Search results for the same pages.

Results

```
Pageranks using Gauss elimination:
A = 22.8856%
B = 29.7112%
C = 23.2028%
D = 22.8856%

Pageranks using inverse matrix multiplication:
A = 22.8856%
B = 29.7112%
C = 23.2028%
D = 22.8856%
```

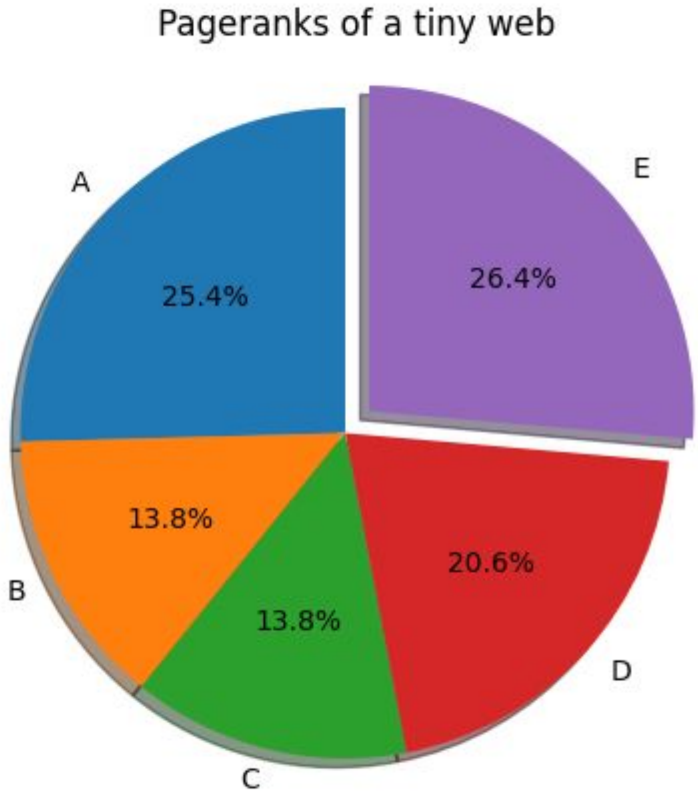
Results



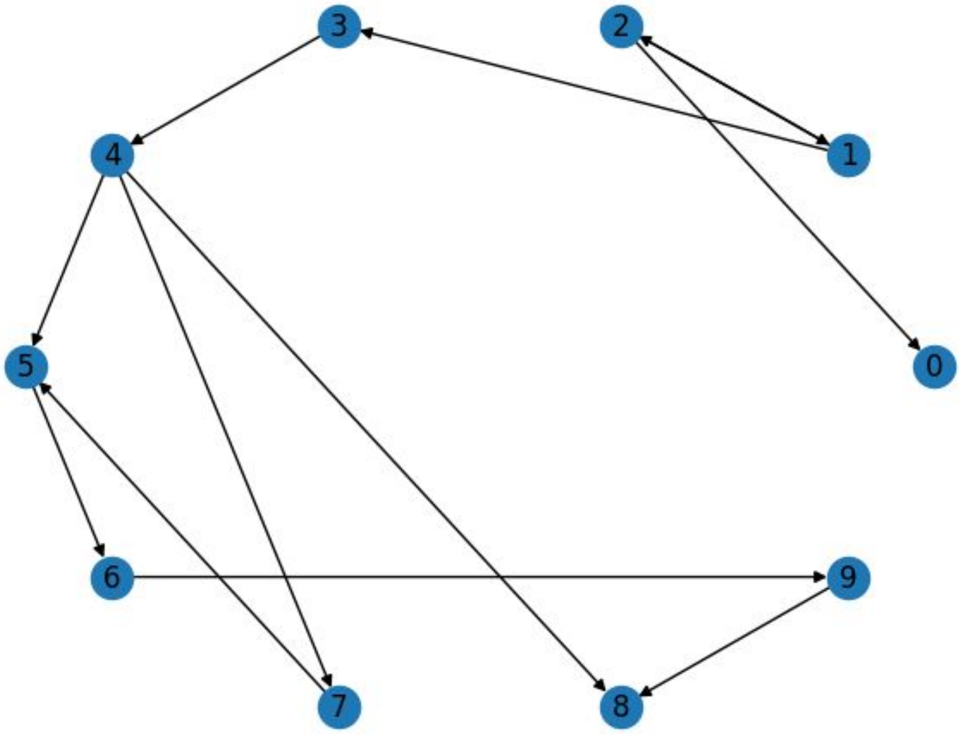
Results

```
Pageranks using power iteration:
A = 25.4192%
B = 13.8032%
C = 13.8032%
D = 20.5990%
E = 26.3755%
```

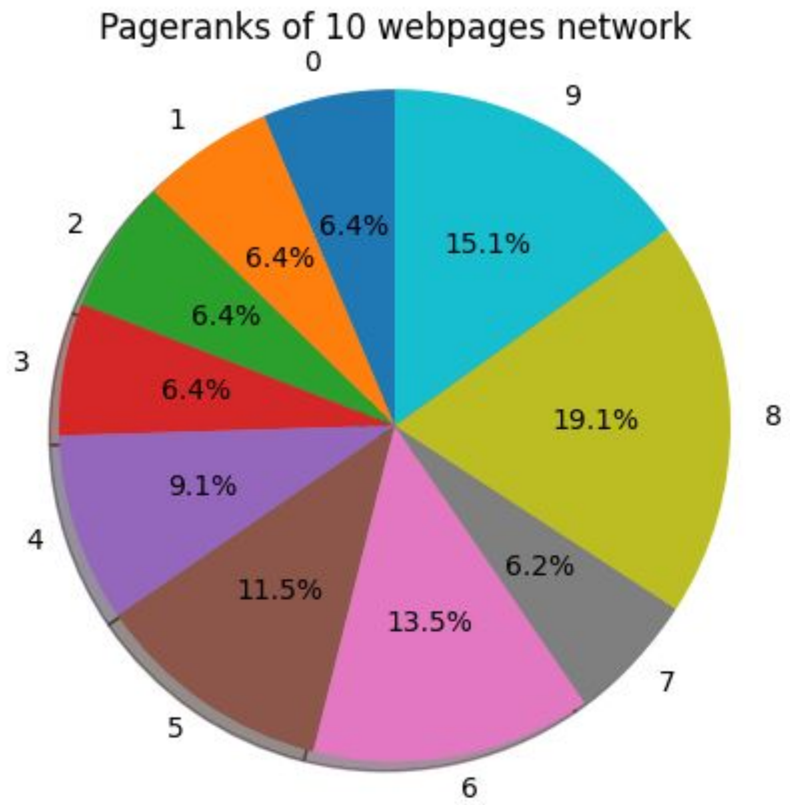
Results



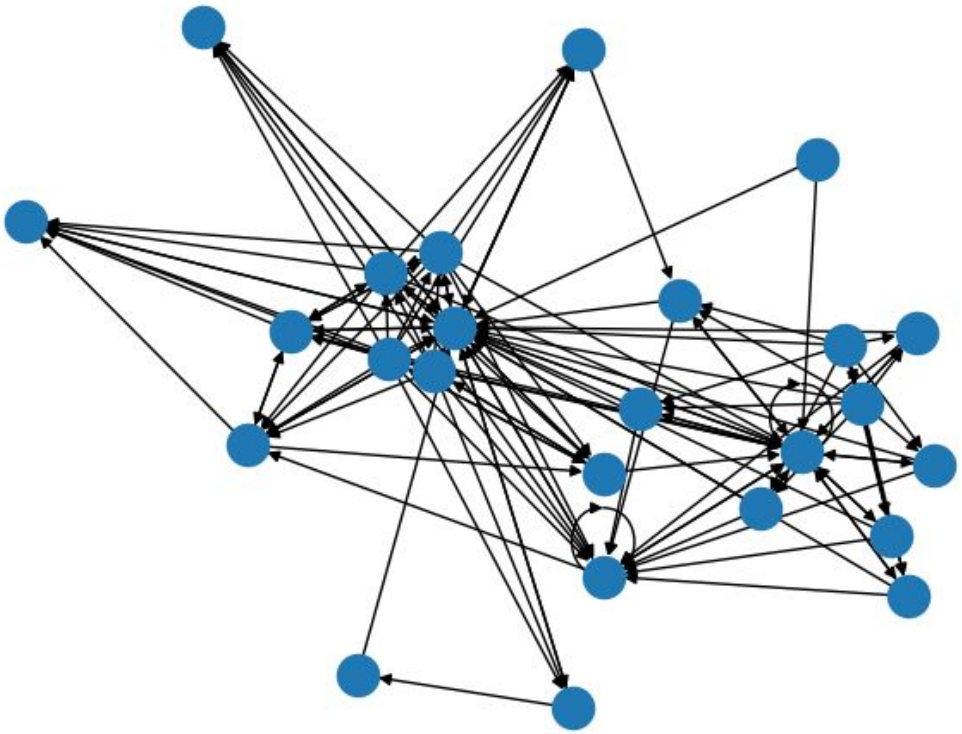
Results



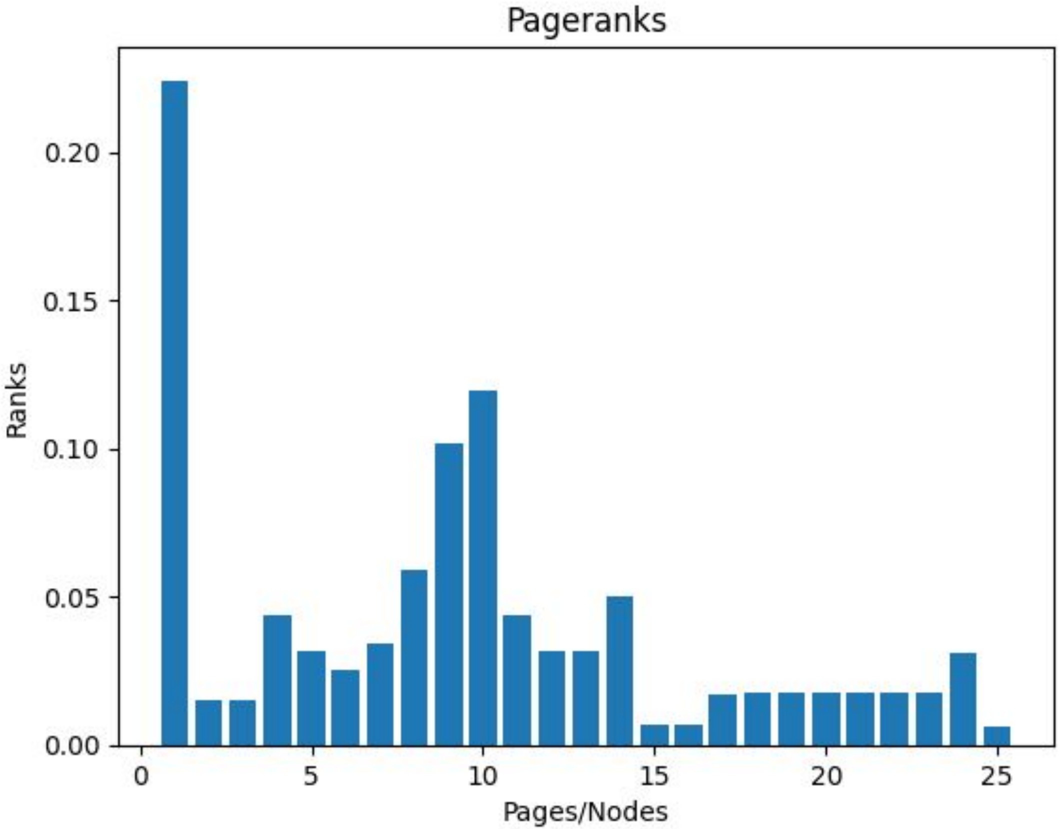
Results


Pageranks of 10 webpages network
0
1
2
3
4
5
6
7
8
9
6.4%
6.4%
6.4%
6.4%
9.1%
11.5%
13.5%
6.2%
19.1%
15.1%

Results



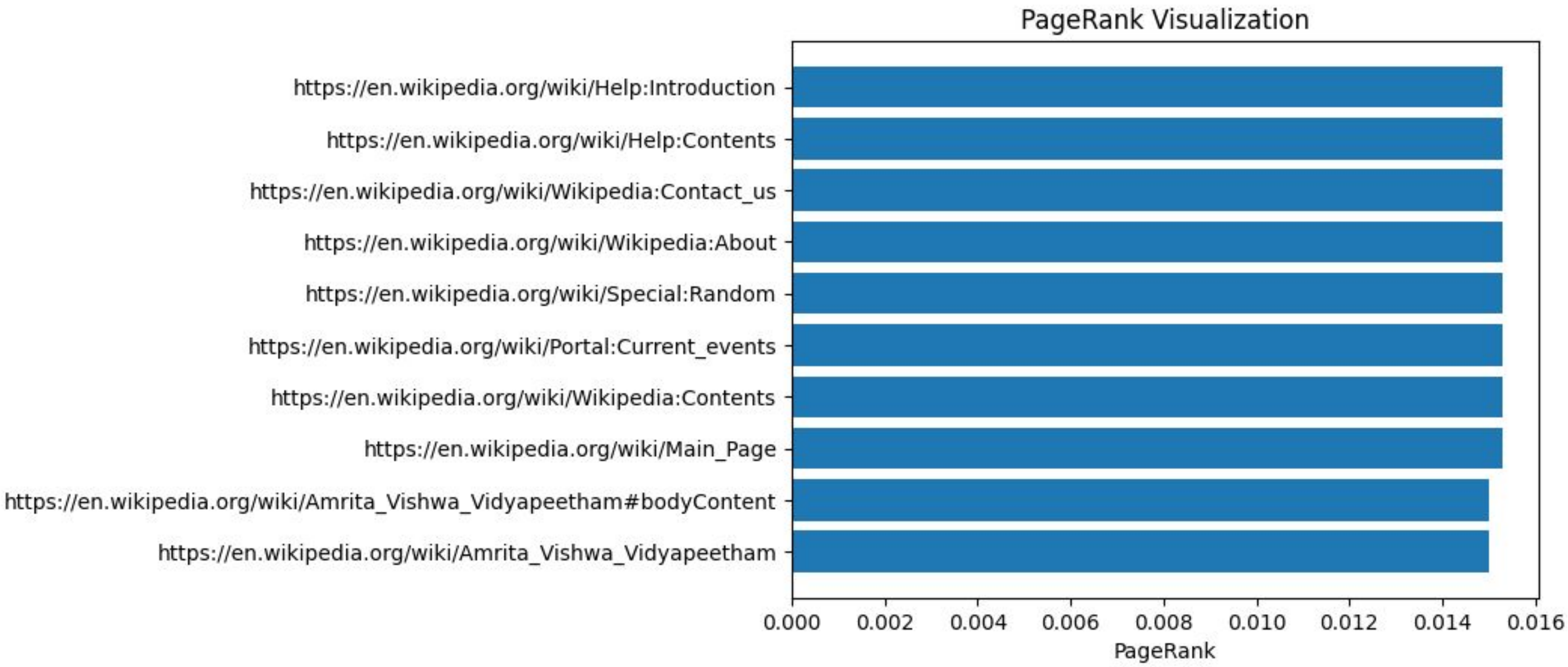
Results



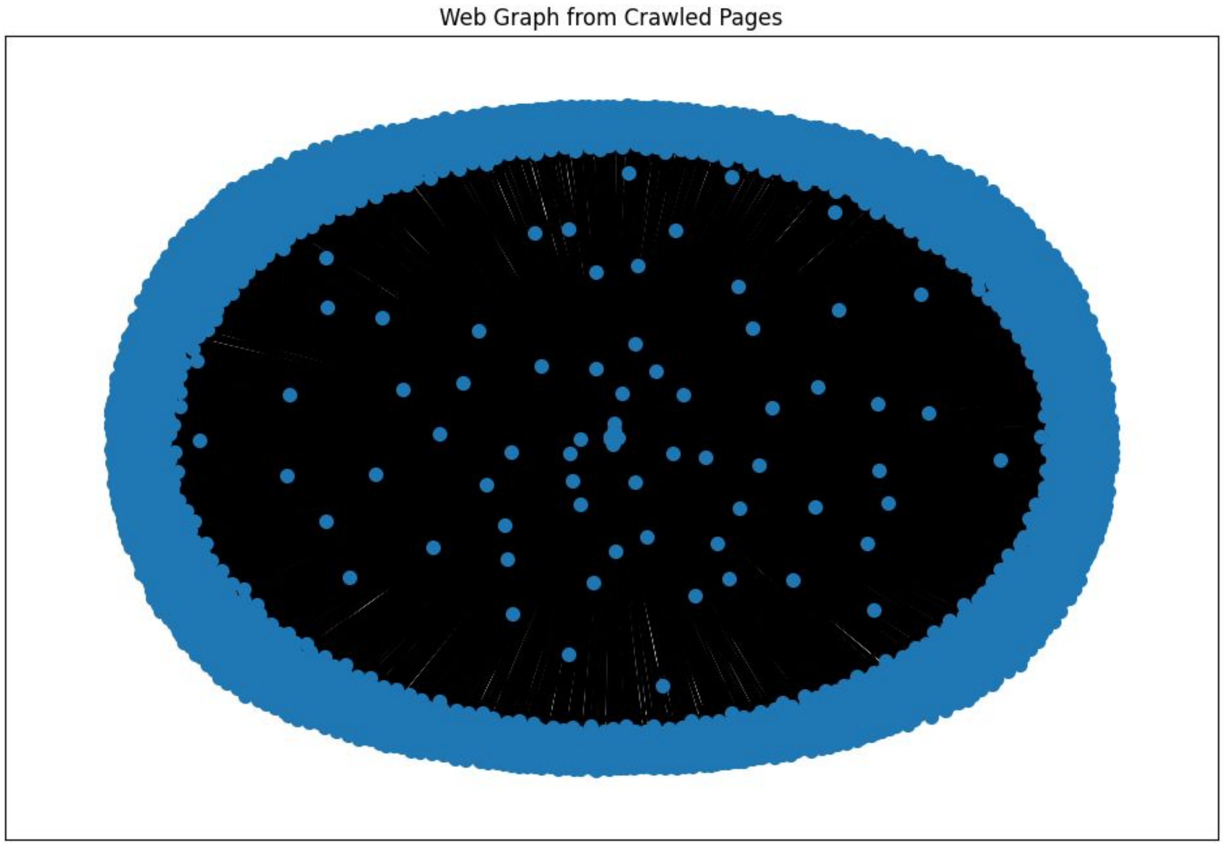
Results

```
PageRank Results:
https://en.wikipedia.org/wiki/Amrita_Vishwa_Vidyapeetham: 0.0150
https://en.wikipedia.org/wiki/Amrita_Vishwa_Vidyapeetham#bodyContent: 0.0150
https://en.wikipedia.org/wiki/Main_Page: 0.0153
https://en.wikipedia.org/wiki/Wikipedia:Contents: 0.0153
https://en.wikipedia.org/wiki/Portal:Current_events: 0.0153
https://en.wikipedia.org/wiki/Special:Random: 0.0153
https://en.wikipedia.org/wiki/Wikipedia:About: 0.0153
https://en.wikipedia.org/wiki/Wikipedia:Contact_us: 0.0153
https://en.wikipedia.org/wiki/Help:Contents: 0.0153
https://en.wikipedia.org/wiki/Help:Introduction: 0.0153
```

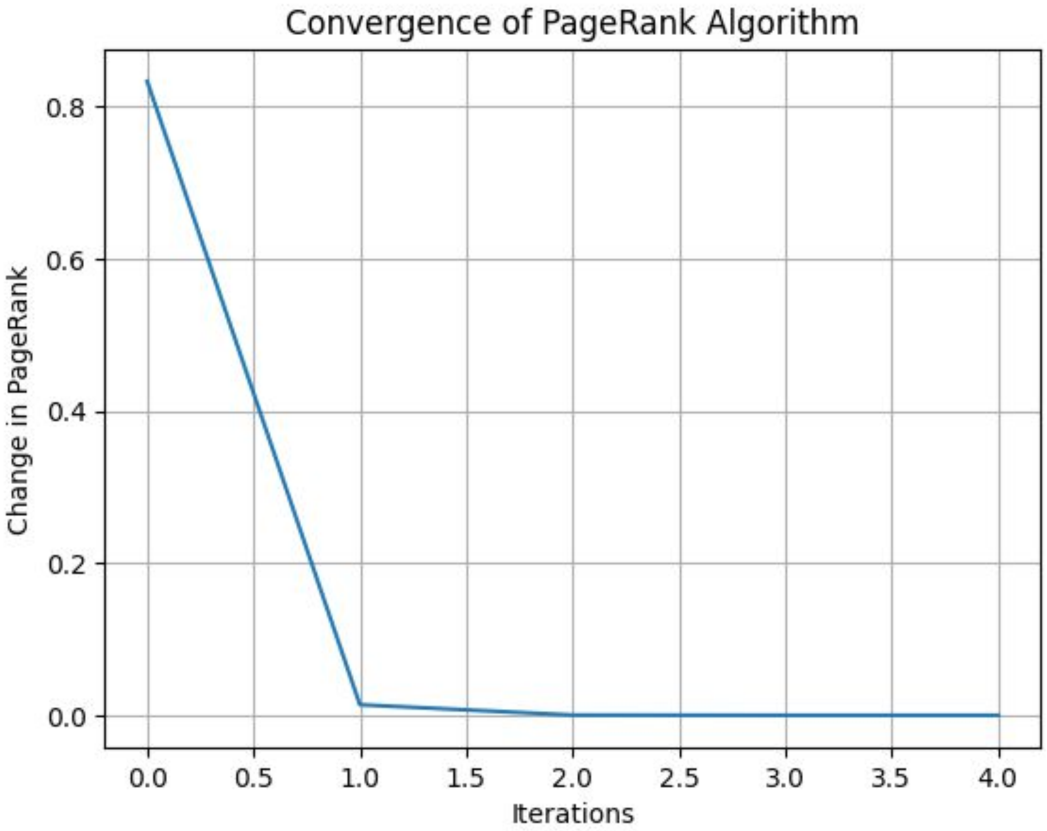
Results


PageRank Visualization
https://en.wikipedia.org/wiki/Help:Introduction
https://en.wikipedia.org/wiki/Help:Contents
https://en.wikipedia.org/wiki/Wikipedia:Contact_us
https://en.wikipedia.org/wiki/Wikipedia:About
https://en.wikipedia.org/wiki/Special:Random
https://en.wikipedia.org/wiki/Portal:Current_events
https://en.wikipedia.org/wiki/Wikipedia:Contents
https://en.wikipedia.org/wiki/Main_Page
https://en.wikipedia.org/wiki/Amrita_Vishwa_Vidyapeetham#bodyContent
https://en.wikipedia.org/wiki/Amrita_Vishwa_Vidyapeetham
0.000
0.002
0.004
0.006
0.008
0.010
0.012
0.014
0.016
PageRank

Results



Results



Convergence of PageRank Algorithm
Change in PageRank
0.0
0.2
0.4
0.6
0.8
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Iterations

Results

```
Top 5 pages by PageRank:
https://en.wikipedia.org/wiki/Main_Page: 0.0153
https://en.wikipedia.org/wiki/Wikipedia:Contents: 0.0153
https://en.wikipedia.org/wiki/Portal:Current_events: 0.0153
https://en.wikipedia.org/wiki/Special:Random: 0.0153
https://en.wikipedia.org/wiki/Wikipedia:About: 0.0153
```

Thank You!