NANYANG TECHNOLOGICAL UNIVERSITY

College of Computing and Data Science (CCDS)

SC4052: Cloud Computing

Academic Year: 2024-2025, Semester 2

Assignment 2 Report

M Hisham B Khairul A (U2121992E)

Contents

Introduction

This assignment touches upon several page-ranking algorithms found in the paper *Stable Algorithms for Analysis* [1], including their implementation and behaviour, and also briefly talks about how LLMs can be integrated.

Graph generation

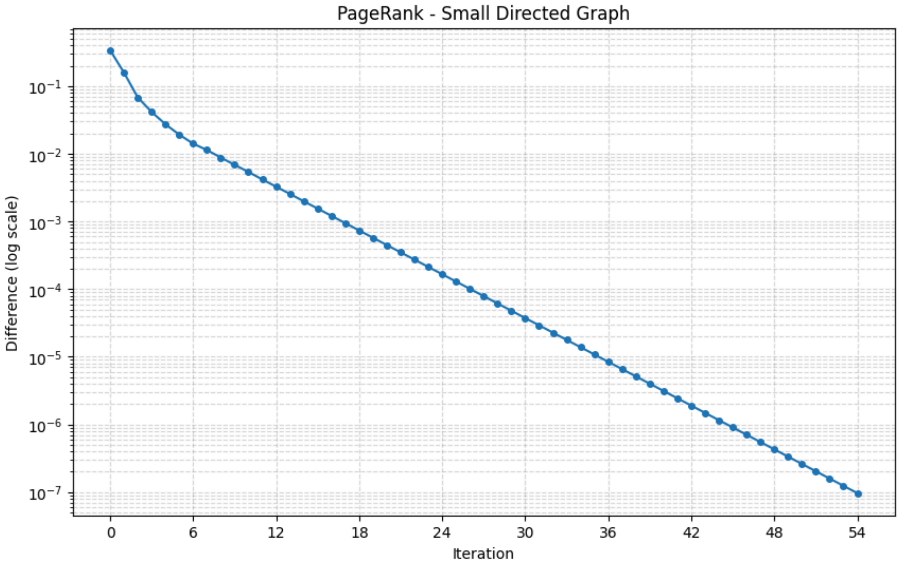
Page-ranking algorithms are used to determine the importance of each page in a set of pages. A page can be boiled down to a node, where a backlink is just an edge pointing to it, and a forward link is just an edge from it to another node. Thus, for each algorithm, we will generate and use 2 directed graphs, 1 small (**15 nodes**) and 1 large (**100 nodes**).

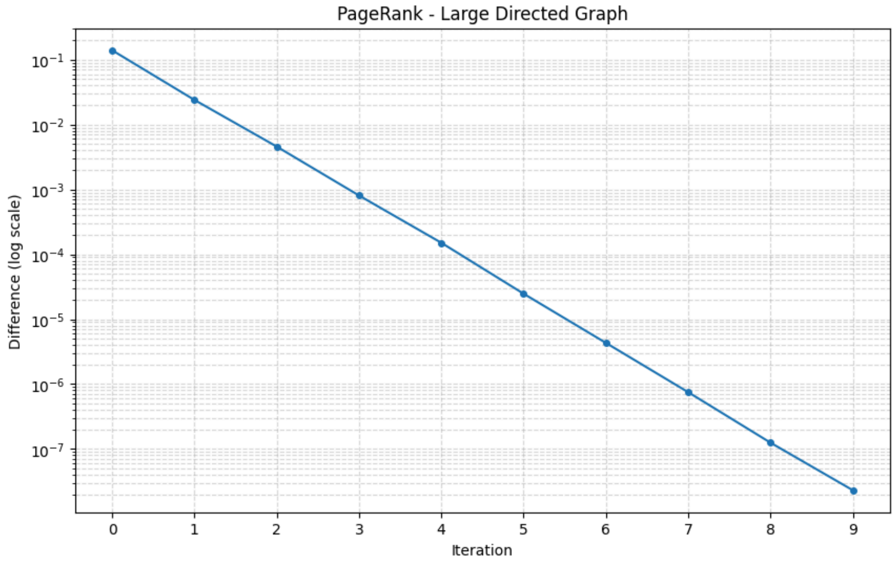
PageRank

The most popular page-ranking algorithm is PageRank. The following formula is used to calculate rank for all nodes at each iteration:

$$PR(p_i) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)}$$

Where d is a “damping factor” to simulate random hops of real users, and L is the number of forward links for that page. We track the total difference of rank values between each iteration, with difference under a given tolerance indicating convergence of the values. We use tolerance of 1e-7:





The small graph converges in 55 iterations, while the large graph converges in 10 iterations. This faster iteration may occur due to stronger structure found in larger graphs, as well as lower rank transference per iteration resulting in "smoother" convergence.

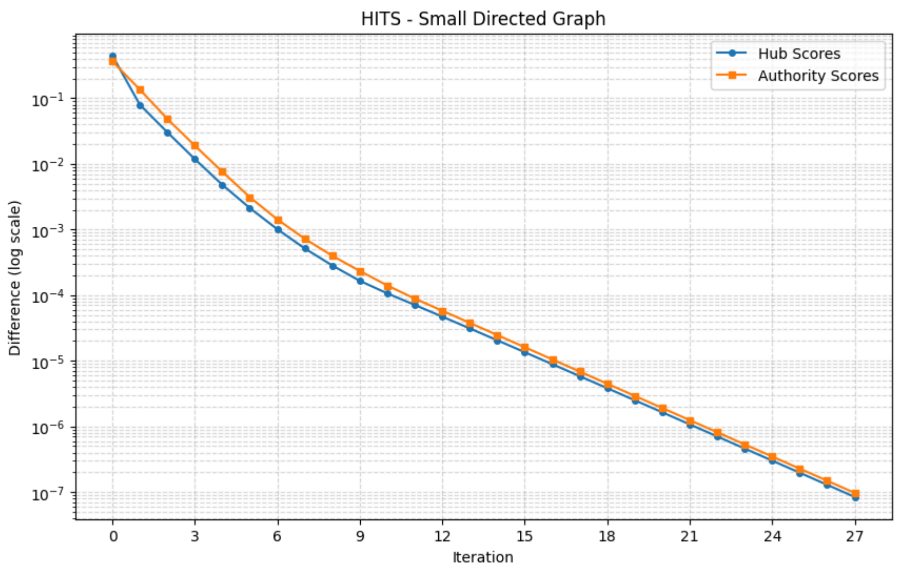
HITS

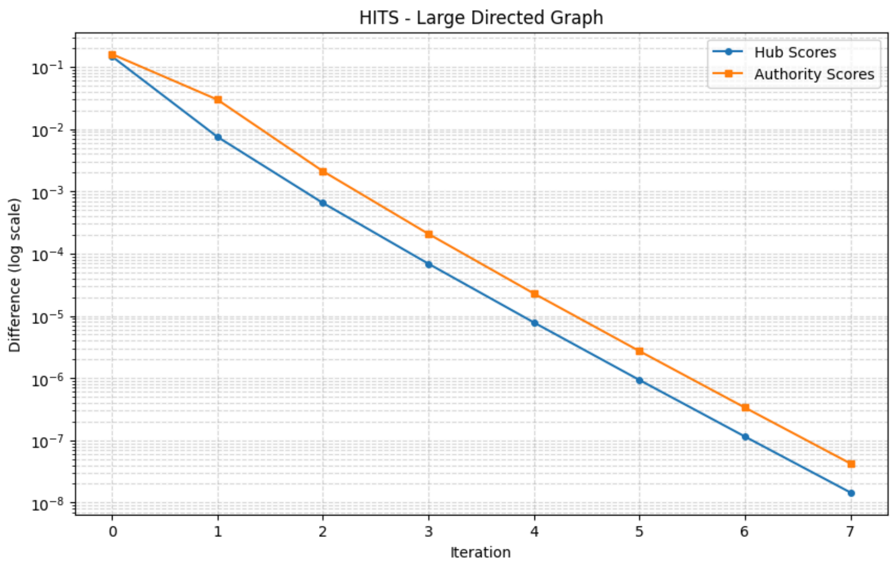
HITS, or Hyperlink-Induced Topic Search, is another page-ranking algorithm. It uses the concepts of "authorities" and "hubs". A strong authority is pointed to by many hubs (highlighting how informative it is), while a strong hub points to many strong authorities (highlighting how it is a good catalog of information). The following 2 formulae are used to calculate these 2 values for each node:

$$Authority(V_i) = \sum_{V_j \in In(V_i)} e_{ji} \cdot Hub(V_j) \qquad (1)$$

$$Hub(V_i) = \sum_{V_j \in Out(V_i)} e_{ij} \cdot Authority(V_j) \qquad (2)$$

Running the algorithm on small and large graphs produces the following convergence results:





The graphs converge at 28 and 8 iterations respectively, showing that HITS may be faster than PageRank, though there is a slightly higher cost per iteration (2 multiplications).

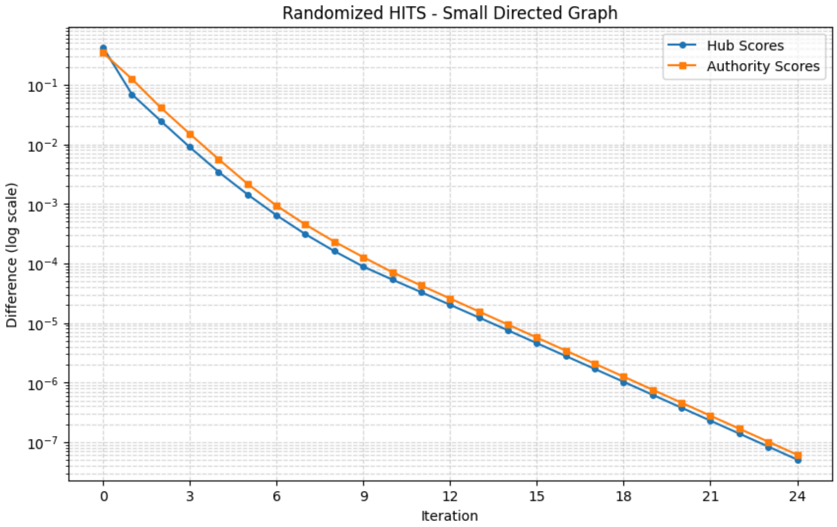
Randomized HITS

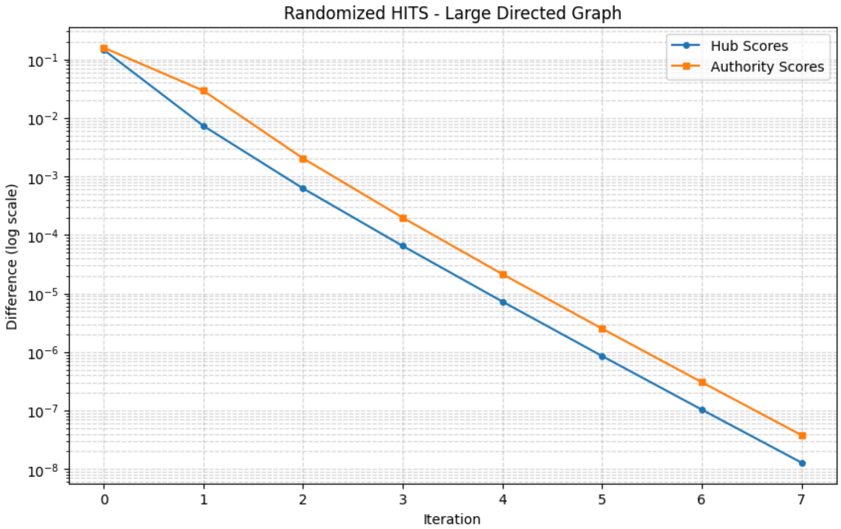
Randomized HITS modifies HITS by using PageRank's random walk methodology. At each step, there is a small chance of teleporting to a random page, like in PageRank. Otherwise, the algorithm alternates between jumping to a random backlink and forward link, counting the number of appearances in both contexts for each node. As the algorithm progresses, the probability of landing at each node as a backlink represents the node's hub score, whilst the probability of landing as a forward link represents its authority score.

The paper provides the following formulae for calculating hub and authority scores as iterations:

$$\begin{aligned} a^{(t+1)} &= \epsilon\vec{1} + (1-\epsilon)A_{\text{row}}^{T} h^{(t)} \\ h^{(t+1)} &= \epsilon\vec{1} + (1-\epsilon)A_{\text{col}} a^{(t+1)} \end{aligned}$$

Implementing this formula gives the following convergence results:



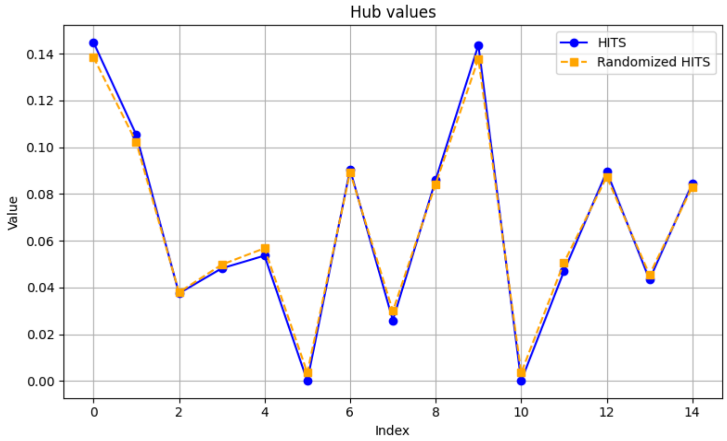


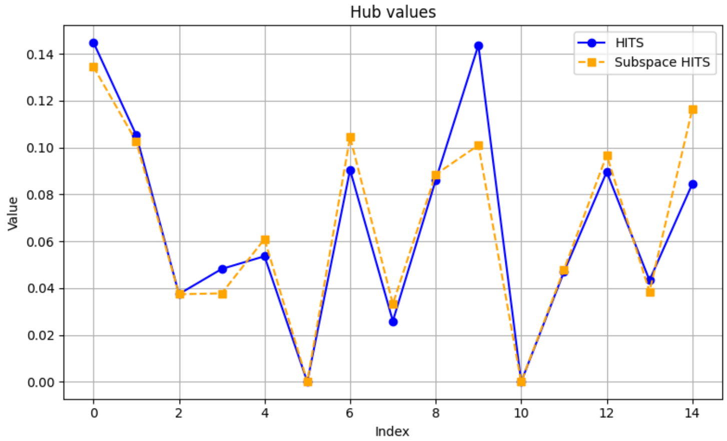
The graphs converge at 25 and 8 iterations respectively, close to normal HITS.

Subspace HITS

Subspace HITS is another extension of HITS. In normal HITS, repeated iteration tends towards the strongest eigenvectors of the graph matrices. Subspace HITS utilizes the top k eigenvectors for the final authority and hub values, which may encode different underlying aspects of the graph model, resulting in a more accurate ranking.

To show what this might look like, we can show the difference between the final hub values for all 3 HITS variants:





We can see that HITS and randomized HITS result in nearly the same values, as they have fairly similar underlying mechanisms. However, Subspace HITS, while close, has some obvious differences which may be a sign of additional structure to the graph model.

Integrating with LLMs

The above algorithms rely solely on links to rank pages. However, in the context of searching, the importance of each page also depends heavily on its relevance to what is being searched for. As such, LLMs can be useful in ingesting the content of each page, and ranking its relevance to a given search term. This LLM-generated ranking can then be used as an additional weight during PageRank/HITS iteration calculations, which may lead to a more accurate final ranking.

References

1. Ng, A. Y., Zheng, A. X., & Jordan, M. I. (2001). *Stable Algorithms for Link Analysis*. https://doi.org/10.1145/383952.384003

Appendix

See attached Jupyter Notebook for code