PART A

(PART A : TO BE REFFERED BY STUDENTS)

Experiment No.10

Aim: Implementation of Page rank / HITS Algorithm.

Outcome:

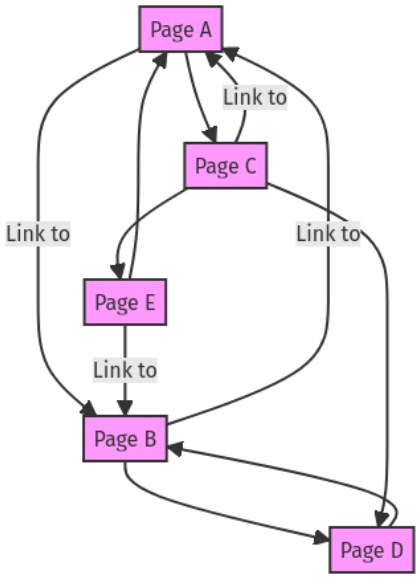
After successful completion of this experiment students will be able to

1. Demonstrate an understanding of the importance of data mining
2. Perform exploratory analysis of the data to be used for mining.

Theory:

The History of Google Page Rank

PageRank was developed by Larry Page and Sergey Brin, the founders of Google, while they were Ph.D. students at Stanford University in the late 1990s. The idea behind PageRank is seemingly simple: it ranks web pages based on their importance, as determined by the number and quality of links to them.



This was a significant departure from existing search engines that ranked results based primarily on how many times a search term appeared on a webpage. PageRank treated the burgeoning internet like a graph, with pages as nodes and hyperlinks as connections between them. Links were essentially a voting system - each link to a page was a vote for its importance - but not all votes were equal. Links from more important pages (i.e., those with more of their own links in) were given more weight than those from less important pages (i.e., those with fewer links into them).

The name "PageRank" is a play on multiple levels: it refers to webpages, but also to Larry Page himself. While the concept started as a primarily academic endeavor, the success of the algorithm and the growing popularity and usefulness of the internet drove Page and Brin to start the company whose name has become synonymous with searching the internet: Google.

Understanding PageRank

Fundamentally, here's how the PageRank calculation works:

Given a defined set of linked web pages, PageRank calculates the probability that a person who clicks any single, random link will end up on a particular page. This allows the algorithm to assign a weight between 0 and 1 that any particular page will be the next one.

PageRank then uses the weight of that particular page (i.e., how likely it is to be clicked) to assign an even weight to each of its *outgoing* links. For example, if a page with a calculated PageRank value of .25 links to two other pages, it would confer half of its PageRank score to each of those pages (.125).

In mathematical terms, a simple version of the PageRank algorithm can be expressed as:

$$PR(u) = \sum_{v \in B_u} \frac{PR(v)}{L(v)}$$

Where the PageRank (PR(u)) of any particular page is the summation of all pages linking to the page's PageRank divided by the number of links each has.

In practice, the actual algorithm Page created also takes into account multiple links from the same page and a damping factor to account for the fact that each link click makes the user less likely to click another link, but I won't go any further into the math here. You can read the entire patent if you're interested.

Once PageRank was calculated for every page in a set of web pages, Google could combine PageRank with the density of a given keyword to calculate a score for each page in a list of search results. For example, if the user searched for "war and peace," Google might find all the pages in its index that mentioned "war and peace" at least five times and then rank them from highest to lowest page rank. The first ten results with the highest PageRank would then be shown to the user as the first page of search results. This example is overly simplistic but illustrates the way that keyword density and PageRank worked together in the early iterations of Google's search engine.

The Evolution of PageRank and Search Rankings

Over time, as Google gained market share and people began to understand PageRank, bad actors began to find ways to take advantage of it. "Black Hat" SEOs built link farms and spam comment bots to generate thousands of artificial signals designed to boost a website's PageRank. These efforts generally didn't reward good content but instead allowed companies to *buy* their way up the rankings.

Google found ways to tamp down these efforts by balancing PageRank with other ranking factors and manual reviews. They continue to change these factors every few weeks or months as new limitations and exploits are discovered. Recent updates have been aimed at thin content, AI-generated content, and non-helpful content.

Google has also stopped being as transparent in its approach to search rankings. Unlike the PageRank algorithm, Google's current ranking methodology has not been filed in a patent or shared publicly, so while Google offers advice and best practices, it's impossible to tell exactly how they'll rank any particular web page.

Checking Your Site's PageRank

Because the original PageRank algorithm was relatively straightforward, Google used to let you check your website's PageRank through its API. This allowed Chrome extensions to annotate search results or show a page's PageRank in real time through the browser. Unfortunately, this API was removed in 2016, so you can't actually determine Google's official PageRank for your site anymore.

That said, there are some alternatives that will approximate your site's PageRank or similar metrics. For example, Ahrefs will give you any site's "Domain Rating," which serves as a measure of your site's authority in relation to all the other sites in Ahrefs' index, and "URL Rating," which does the same for a single URL of a site.

Semrush, Moz, and Positional all offer similar metrics to help you understand how your page or domain compares with others. While these numbers are *not* exactly the same as Google's official PageRank, they're useful in benchmarking performance and understanding the effectiveness of your SEO efforts. PageRank is also often described as domain authority, "link juice," and "link equity."

Finally, it's important to note that PageRank is not as big a part of Google's core algorithm anymore. While checking your site's rating using third-party tools is a good idea, you should also understand all the factors that go into Google rankings and some of the ways you can positively impact them.

Improving Your Site's Rankings and PageRank in 2024

While PageRank isn't as big a factor in Google's search rankings anymore, it still holds some fundamental value. Google still uses reputable links as a factor in rankings, but they also use links to help them understand the relevance and relationship of pages to one another. So, improving your PageRank by increasing the number of high-quality inbound links to your site is generally a good thing, but it's not the *only* thing anymore. Additionally, you have to be careful about paying for links or getting low-quality links, as these could do more harm than good.

Despite PageRank's diminishing role in search results, there are many things you can do to help your site rank higher in Google search rankings. Here are a few of the factors you can work on immediately to improve your search rankings:

1. **User Experience** - Focus on usability (including Core Web Vitals) as this is an increasingly important part of search rankings. Users want a site that is fast, easy to navigate, and error-free.
2. **Keyword Optimization** - Use relevant, high-volume, low-difficulty keywords that your target audience is searching for. Integrate these keywords naturally into your website's content, titles, meta descriptions, and URLs. The goal is to make your content's topic more clear to crawlers while avoiding keyword stuffing.
3. **Content Quality, Freshness, and Multimedia** - Prioritize high-quality, original, and informative content that addresses your audience's needs. Regularly update content to keep it fresh and incorporate diverse media formats like video, images, and diagrams when relevant.
4. **Social Signals** - While a large social following might not directly improve your rankings, building an audience will help you generate more backlinks by encouraging others to share your content. Also, Google has gotten increasingly good at understanding brand signals on other platforms, for example, TikTok, and using those signals to influence organic search results.
5. **Mobile Responsiveness and Accessibility** - Ensure your website is optimized for mobile users and those using screen readers or assistive devices. Responsive designs, proper alt tags on images, fast loading times, and easy navigation on smaller screens are all important considerations.

6. **Technical SEO and Site Structure** - Ensure that structured data is properly displayed, a secure (HTTPS) connection is used, XML sitemaps are updated, and that you're using internal links effectively.
7. **Local SEO** - For businesses targeting a particular local market (e.g., restaurants, storefronts, event venues, etc.), optimizing for local SEO is crucial. Use local keywords, maintain listings on your Google Business Profile, encourage customer reviews, and create location-specific content.

PART B

(PART B : TO BE COMPLETED BY STUDENTS)

(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded at the end of the practical)

Roll. No: A57	Name: RIYA PRASANNA DEVADIGA
Class: T.E(AI&DS)	Batch: A3
Date of Experiment: 07/10/2024	Date of Submission: 07/10/2024
Grade:	

Input and Output:
PAGE- RANK ALGORITHM:

```
import numpy as np

def pagerank(links, num_iterations=100, d=0.85):
    num_pages = len(links)
    # Initialize page ranks to 1/n
    ranks = np.ones(num_pages) / num_pages

    for _ in range(num_iterations):
        new_ranks = np.zeros(num_pages)
        for i in range(num_pages):
            for j in range(num_pages):
                if links[j][i] == 1:  # If there's a link from j to i
                    new_ranks[i] += ranks[j] / np.sum(links[j])  # Distribute rank
        ranks = (1 - d) / num_pages + d * new_ranks  # Apply damping factor

    return ranks

# Example link structure (adjacency matrix)
# 0 -> 1, 1 -> 2, 2 -> 0, 2 -> 1
links = np.array([[0, 1, 0],
                  [0, 0, 1],
                  [1, 1, 0]])
```

```
ranks = pagerank(links)
print("PageRank Scores:", ranks)
```

OUTPUT:

```
[Running] python -u "C:\Users\admin\AppData\Local\Temp\tempCodeRunnerFile.python"
PageRank Scores: [0.21481063 0.39739966 0.38778971]

[Done] exited with code=0 in 0.322 seconds
```

PAGE-HIT ALGORITHM:

```
import numpy as np

def hits(links, num_iterations=100):
    num_pages = len(links)
    authority_scores = np.ones(num_pages)
    hub_scores = np.ones(num_pages)

    for _ in range(num_iterations):
        # Update authority scores
        authority_scores = np.dot(links.T, hub_scores)
        authority_scores /= np.linalg.norm(authority_scores, 2)  # Normalize

        # Update hub scores
        hub_scores = np.dot(links, authority_scores)
        hub_scores /= np.linalg.norm(hub_scores, 2)  # Normalize

    return authority_scores, hub_scores

# Example link structure (adjacency matrix)
links = np.array([[0, 1, 0],
                  [0, 0, 1],
                  [1, 1, 0]])

authority, hub = hits(links)
print("Authority Scores:", authority)
print("Hub Scores:", hub)
```

Output:

```
[Running] python -u "C:\Users\admin\AppData\Local\Temp\tempCodeRunnerFile.python"
Authority Scores: [5.25731112e-01 8.50650808e-01 1.87378877e-42]
Hub Scores: [5.25731112e-01 1.15806515e-42 8.50650808e-01]

[Done] exited with code=0 in 0.204 seconds
```

Observations and learning:

- **Link Analysis**: Both PageRank and HITS are foundational algorithms used for link analysis in networks, especially in the context of the web, where they help rank pages based on their importance.
- **PageRank Mechanism**: PageRank operates on the principle that important pages are likely to be linked to by other important pages. It uses a damping factor to simulate a user randomly surfing the web.
- **HITS Components**: HITS distinguishes between authority and hub scores. Authorities are valuable pages linked to by many hubs, while hubs are pages that link to many authorities, highlighting different aspects of web structure.
- **Iterative Calculation**: Both algorithms rely on iterative updates of scores. This iterative nature means they can converge to stable rankings over time, although they may require many iterations for larger graphs.
- **Applications Beyond Web**: While originally developed for web page ranking, these algorithms can also be applied in social networks, citation analysis, and recommendation systems, showcasing their versatility in various fields.

Conclusion:

PageRank and HITS are seminal algorithms in the field of network analysis, primarily used for ranking web pages based on their importance and relevance. PageRank leverages the concept of link authority, where the significance of a page is determined by the quality and quantity of inbound links, while HITS focuses on distinguishing between hub and authority scores to capture different aspects of web structure.

These algorithms have demonstrated their effectiveness in various applications beyond web ranking, including social network analysis, recommendation systems, and citation networks. Despite their powerful capabilities, they also come with limitations, such as sensitivity to link structure and the need for substantial computational resources in large datasets.

Question of Curiosity:

1. What is the primary purpose of the PageRank algorithm?

A: The primary purpose of the PageRank algorithm is to rank web pages based on their importance and relevance, determined by the quantity and quality of links to them.

2. How does HITS differentiate between hubs and authorities?

A: HITS differentiates between hubs and authorities by assigning hub scores to pages that link to many authoritative pages and authority scores to pages that are linked to by many hubs.

3. What role does the damping factor play in the PageRank algorithm?

A: The damping factor in PageRank simulates the behavior of a user randomly surfing the web, allowing for a certain probability of jumping to any page, which helps stabilize the ranking.

****