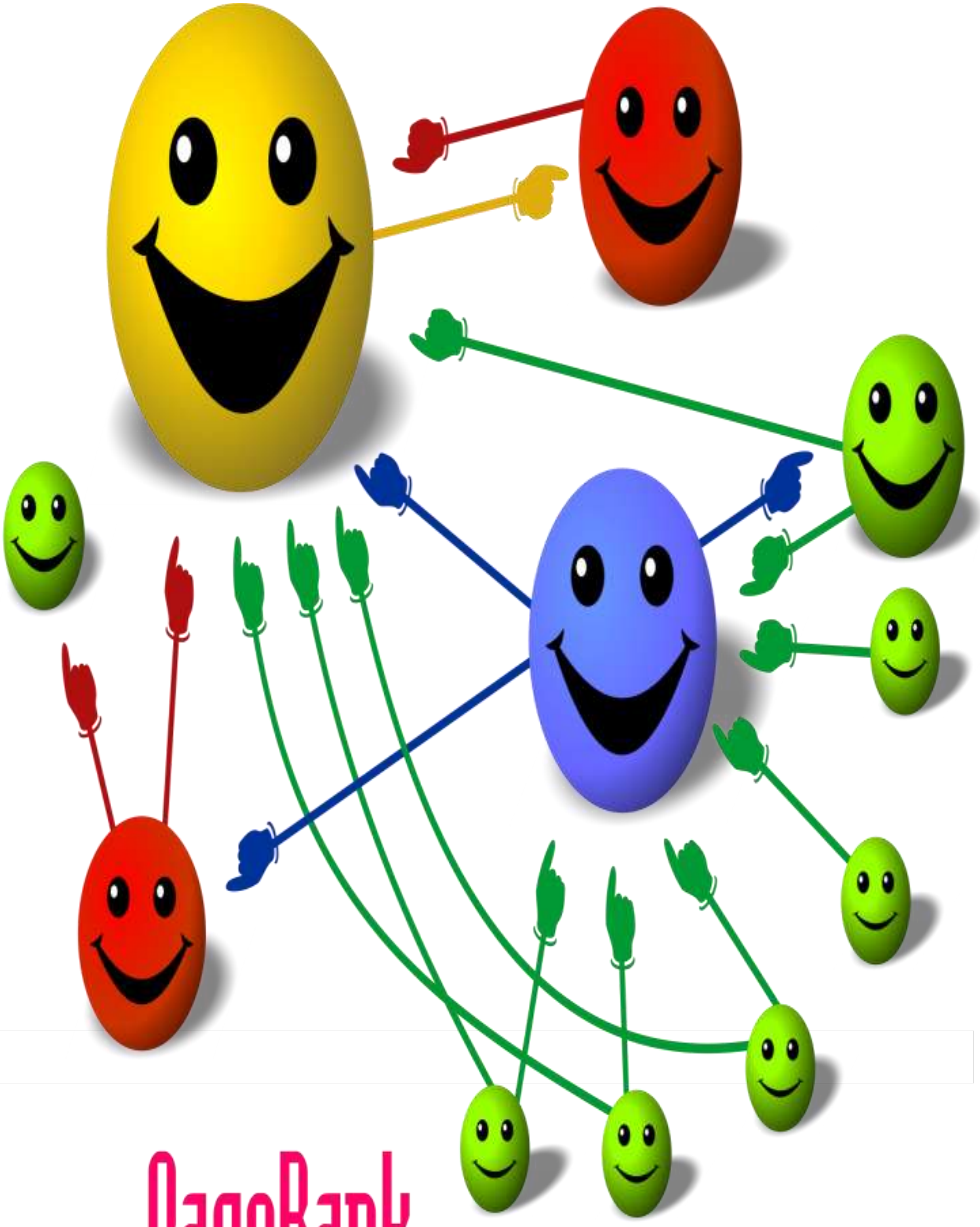

PageRank

جامعة الإسكندرية
ALEXANDRIA
UNIVERSITY

Team Members :

Name	*ID*
Salma Fahmy Hassan	22010346
Rahma Ramadan Hassan	22010334
Marihan Emad Eldeen Mahmoud	22011531
Salma Tarek Abd Elmaged Mohamed	22011629
Salsabil mohamed	22010110
Rowan Kamal Fayad	22010340

1.Definitions:

Markov chain :

A Markov chain is a mathematical model where transitions between states occur based on probabilistic rules. Its key feature is that future states are determined solely by the current state, regardless of the path taken to reach it. This property, known as the "memory-less" property, distinguishes Markov chains from other stochastic processes.

PageRank:

PageRank is a Google algorithm that evaluates the significance of webpages by assessing the quality and quantity of inbound links. It treats incoming links as votes, with pages accruing more high-quality links being considered more influential in search rankings. Essentially, PageRank facilitates higher rankings for websites by leveraging the concept that a page gains importance when it is linked to by other important pages

2.Introduction to web page ranking

Search engines primarily ranked web pages based on simple metrics like keyword frequency and metadata. However, this approach often led to low-quality search results and was susceptible to manipulation by website owners.

PageRank, developed by Larry Page and Sergey Brin at Google, aimed to address these shortcomings by considering the web's structure. It assigns a score to each page based on the number and quality of links pointing to it, with more influential pages receiving higher scores.

The aims of our project are to implement and analyze the PageRank algorithm, considering both sampling and iterative methods. By exploring these methods, we seek to understand their effectiveness in estimating PageRank values and their practical implications for web search algorithms.

3.Methods:

Sampling Method:

- This method relies on randomly selecting pages based on a model that determines the likelihood of each page being chosen. It simulates the behavior of a random surfer moving through web pages.
- To estimate PageRank scores, we repeatedly select pages and record their frequencies. Over time, these frequencies converge to represent the actual PageRank values.

Iterative Method:

- The iterative method continuously updates PageRank values until they stabilize. It starts with initial estimates for each page's PageRank and then refines these estimates through multiple iterations.
- During each iteration, PageRank values are adjusted based on the probabilities of transitioning from one page to another. This process repeats until the PageRank values stop changing significantly, indicating convergence.

Equations:

1. Sampling Method Equation:

$PR(p)=N(p)NPR(p)=NN(p)$

Where:

- $PR(p)PR(p)$ is the PageRank score of page pp.
- $N(p)N(p)$ is the number of times page pp was visited in the sampling process.
- NN is the total number of samples.

2. Iterative Method Equation:

$PR(p)=(1-d)\times 1N+d\times\sum i\in L(p)PR(i)C(i)PR(p)=(1-d)\times N1+d\times\sum i\in L(p)C(i)PR(i)$

Where:

- $PR(p)PR(p)$ is the PageRank of page pp.
- dd is the damping factor.
- NN is the total number of pages in the corpus.
- $L(p)L(p)$ is the set of pages linking to pp.
- $C(i)C(i)$ is the number of outgoing links from page ii.

4.Implementation and practical challenges:

Implementation:

We implemented the PageRank algorithm in Python, utilizing data structures and algorithms to represent web pages, links, and PageRank scores. We used libraries such as `numpy` for numerical computations and `re` for parsing HTML to extract links. The implementation calculates PageRank using both sampling and iterative methods.

a. Imports:

- Describe the imported libraries/modules and their roles in the script.
The script imports essential modules like os, random, re (for regular expressions), sys, and numpy.

b. Constants:

- Explain the significance of DAMPING and SAMPLES constants in the context of the PageRank algorithm.
-DAMPING: A fixed value (typically 0.85) representing the likelihood of a user clicking on links rather than navigating to a new page.
-SAMPLES: The number of iterations used in the sampling variant of the PageRank algorithm.

c. Main Function:

- Provide an overview of the main function's tasks, such as argument validation, calling the crawl function, computing PageRank, and printing results.
Checks if the correct number of command-line arguments is provided.
Calls the crawl function to parse HTML files in the provided directory.
Computes PageRank using both sampling and iteration methods.
Prints the PageRank results

d. Crawl Function:

- Describe how the crawl function parses HTML files to identify links between pages and constructs a corpus representation.

Parses HTML files within a directory to identify inter-page links.

Constructs a dictionary where each page serves as a key, and its associated value is a set of linked pages within the corpus.

e. Transition Model Function:

- Explain how the transition_model function calculates the probability distribution for transitioning between pages.

Furnishes a probability distribution dictating the next page to visit, given the current page.

With a chance represented by damping_factor, it randomly selects a link connected to the current page.

Alternatively, it chooses a link from all pages in the corpus.

f. Sample PageRank Function:

- Detail the process by which the sample_pagerank function estimates PageRank values through random sampling.

. Estimates PageRank values via sampling based on the transition model.

. Commences with a randomly selected page and iterates n times, choosing subsequent pages in accordance with transition model probabilities.

. Yields a dictionary with page names as keys and their estimated PageRank values

g. Iterative PageRank Function:

- Describe the iterative approach used by the iterate_pagerank function to estimate PageRank values until convergence.

. Determines PageRank values iteratively by updating them until convergence.

. Initializes PageRank values for each page and adjusts them until stability is achieved.

. Produces a dictionary mapping page names to their calculated PageRank values.

Script Execution Check:

Ensures execution of the main function if the script is run directly.

Practical Challenges:

One significant challenge was dealing with large-scale web graphs, where the number of web pages and links can be massive. Efficient data structures and algorithms were crucial to handle such large datasets. Additionally, ensuring the convergence and accuracy of PageRank values required careful tuning of parameters and iterative refinement.

5.conclusions :

In conclusion, the PageRank algorithm has significantly improved the quality of web search results by considering the structure and connectivity of the web. By analyzing links between pages, PageRank provides more accurate and relevant search rankings, benefiting both users and website owners.

Through our implementation and analysis, we've gained a deeper understanding of PageRank and its applications in web search. The project successfully demonstrates the effectiveness of both sampling and iterative methods in estimating PageRank values and highlights the challenges associated with processing large-scale web graphs.