





# Table of Contents

Overview of the World Wide Web (WWW)

Architecture of the Google Search Engine

Link Analysis Techniques: PageRank and HITS



# The World Wide Web (WWW)

- **Definition**

- *“All Internet resources and users using the Hypertext Transfer Protocol (HTTP)”*

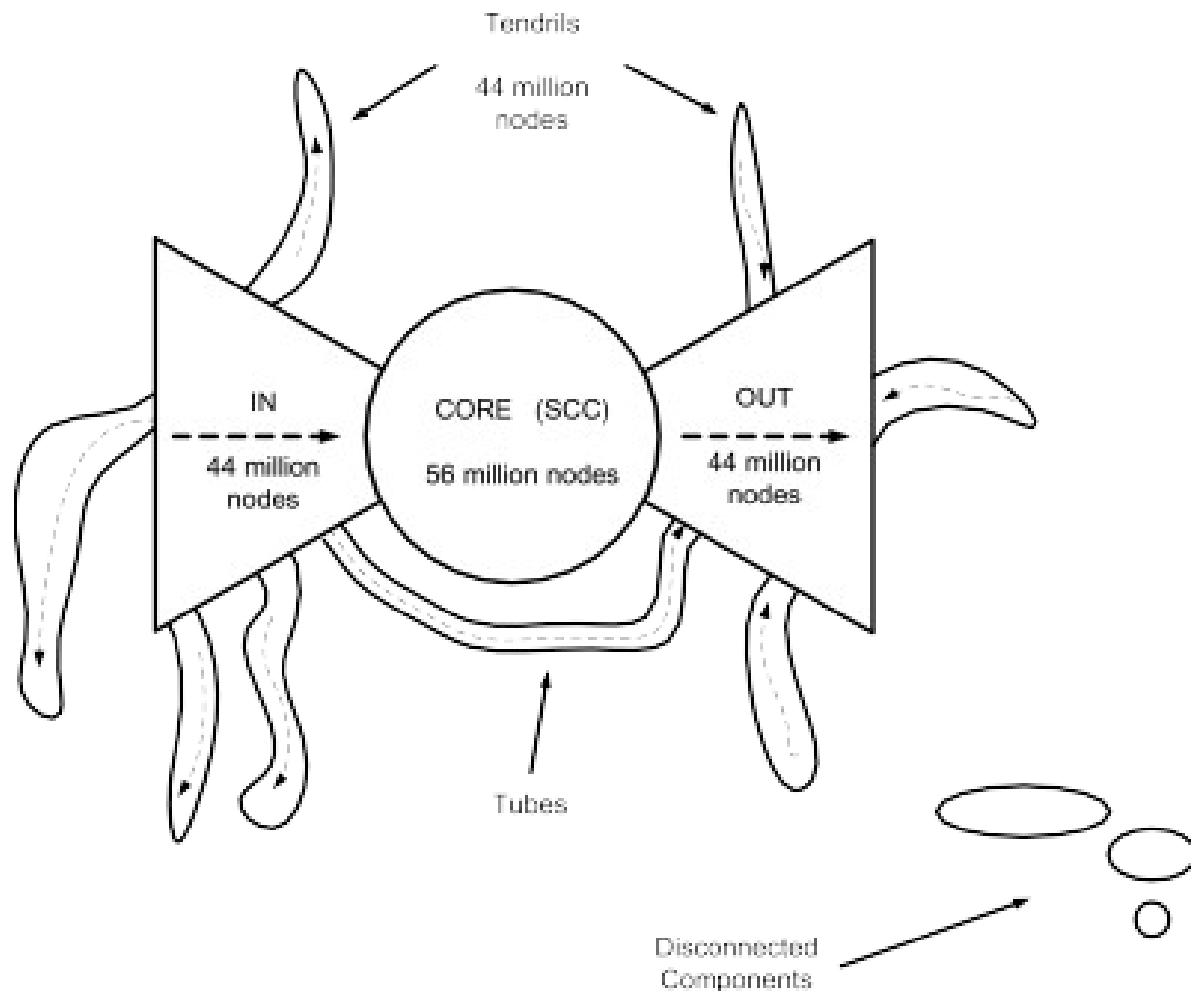
- **More general definition** (Sir Tim Berners-Lee):

- *“The World Wide Web is the universe of network- accessible information, an embodiment of human knowledge”*
-



# With **Anything** And **Everything** Present

The web is a messy place



## Web Graph Structure: The Bow-Tie Mode

- **CORE (SCC):**  
Strongly Connected Component where every page can reach every other page.
- **IN:**  
Pages that link *to* the CORE but can't be reached *from* it.
- **OUT:**  
Pages that can be reached *from* the CORE but don't link back.
- **Tendrils:**  
Pages not connected to the CORE — can't reach it and can't be reached from it.
- **Tubes:**  
Pages that connect IN to OUT without passing through the CORE.



# Web Search Taxonomy

- **Navigational**
  - Aimed at reaching a specific website  
*e.g., “Facebook login”, “YouTube”*
- **Informational**
  - Seeking information available online  
*e.g., “symptoms of flu”, “how does blockchain work”*
- **Transactional**
  - Intending to perform an action or transaction  
*e.g., “buy iPhone 13”, “book flight to NYC”*
- **Resource**
  - Looking to access a tool or file  
*e.g., “PDF to Word converter”, “download Python”*



# Challenges in Web Information Retrieval

## **Too Much Info**

- The web is growing faster than we can search it.

## **Dead Links (404s)**

- Pages vanish, move, or become outdated.

## **Mixed Formats**

- Text, video, images, PDFs—hard to process consistently.

## **Unreliable Quality**


- Anyone can publish—credibility varies.

## **Language Barriers**

- Valuable content may be locked in other languages.



How can we **improve** content discovery over **traditional search**?

 **Browsing:** Navigate through **hierarchical categories** for more specific and relevant web content, as seen in directories like Yahoo! and DMOZ.





# Simplifying Document Search with Categories



Browsing by **relevant categories** makes finding documents easier.



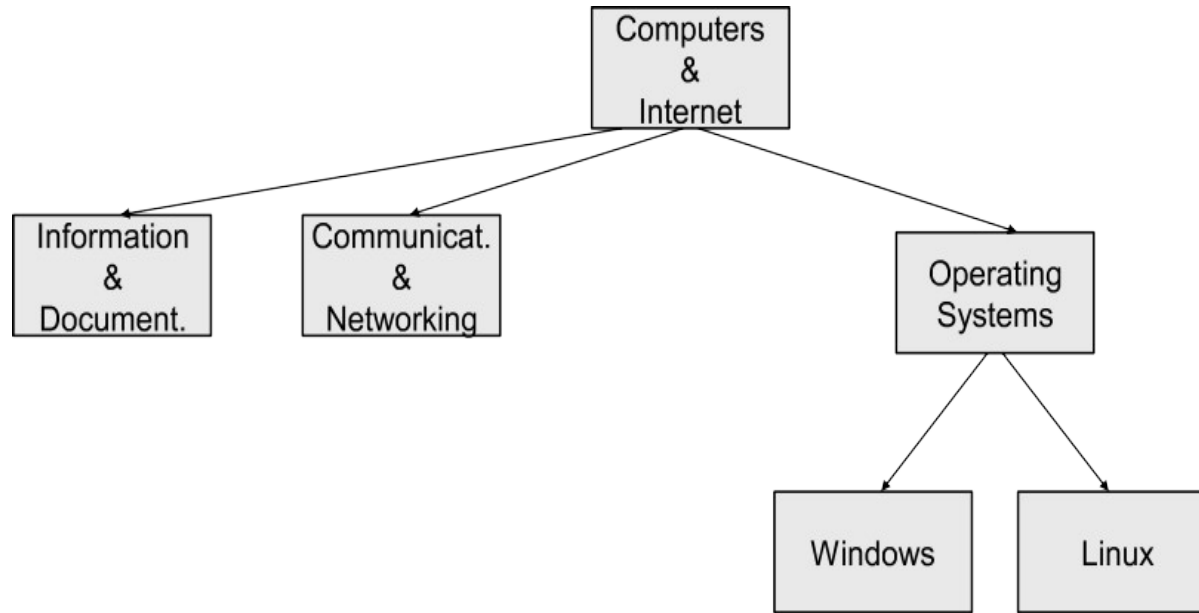
Catalogues like **Yahoo! Directory** and **DMOZ** (now defunct) used a **hierarchical structure** to organize web content.



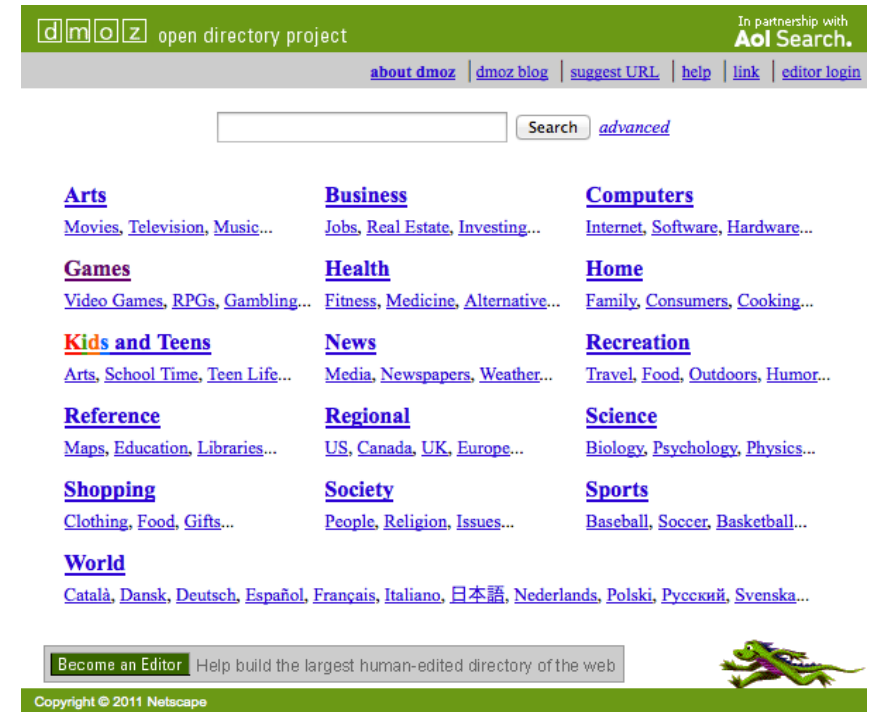
Each document was **tagged into one or more categories**, offering valuable context about its topic.



As you **drill deeper into the hierarchy**, topics become **more specific and focused**.



**Hierarchical Category Scheme Example**  
***Yahoo! Computers & Internet***





# Browsing: Advantages & Disadvantages

## Advantages

- **Higher Precision:** Search space is confined, leading to more accurate results.
- **Handles Homonyms/Polysemes:** Searching “tree” in “Forestry” filters out irrelevant results.
- **More Relevant Documents:** Discover related content by exploring the same category.
- **No Query Needed:** Users can browse without formulating a search query.
- **No Deep Knowledge Required:** Ideal for users unfamiliar with the topic.

## Disadvantages

- **Pre-knowledge Required:** Users must know the category to visit.
- **Uncategorized Documents:** Many documents remain uncategorized.



**Browsing** is about exploring content casually, but what if I am looking for a specific information in the Web?






We **SEARCH !**

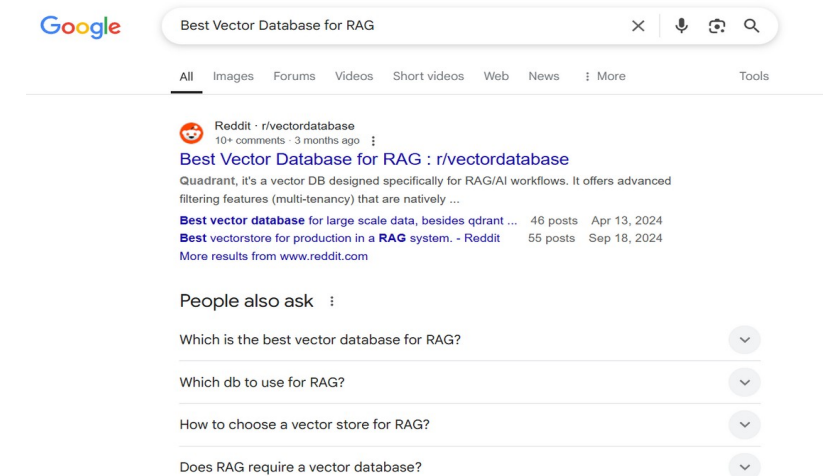


# Searching the web

Searching is a focused approach where users actively look for specific information or answers, leveraging keywords and algorithms to quickly retrieve relevant results.

The vast amount of Web documents cannot be made available without **search technology** which includes

-  **Crawlers & Bots:** Explore and gather content from the web. ✓
-  **Databases & Indexes:** Store and organize indexed web content for quick retrieval. ✓
-  **Natural Language Processing (NLP):** Understand and interpret user search queries. ✓
-  **Machine Learning:** Continuously improve search result relevance. ✓
-  **Ranking Algorithms:** Algorithms like Google PageRank and HIT to rank search results based on relevance. 🚫➡️



We will be exploring a Ranking Algorithm in the upcoming slide, but before that lets understand the anatomy of **THE WEB**


# More on the technical Anatomy of the Search Engine


*Understanding the Early*





 **Crawler** : Visits websites & collects web pages.


## That's an example of Unix based WEB CRAWLER


 **URL Server** : Collects URLs from the index & sends them to the crawler.


 **URL Resolver**: Reads anchors, converts relative → absolute URLs & doc IDs, and builds the links database.


 **Indexer**: Parses pages into HITS (terms & links), builds forward/inverted indexes & updates lexicon.


 **PageRank**: Computes page ranks using the links database's structure.

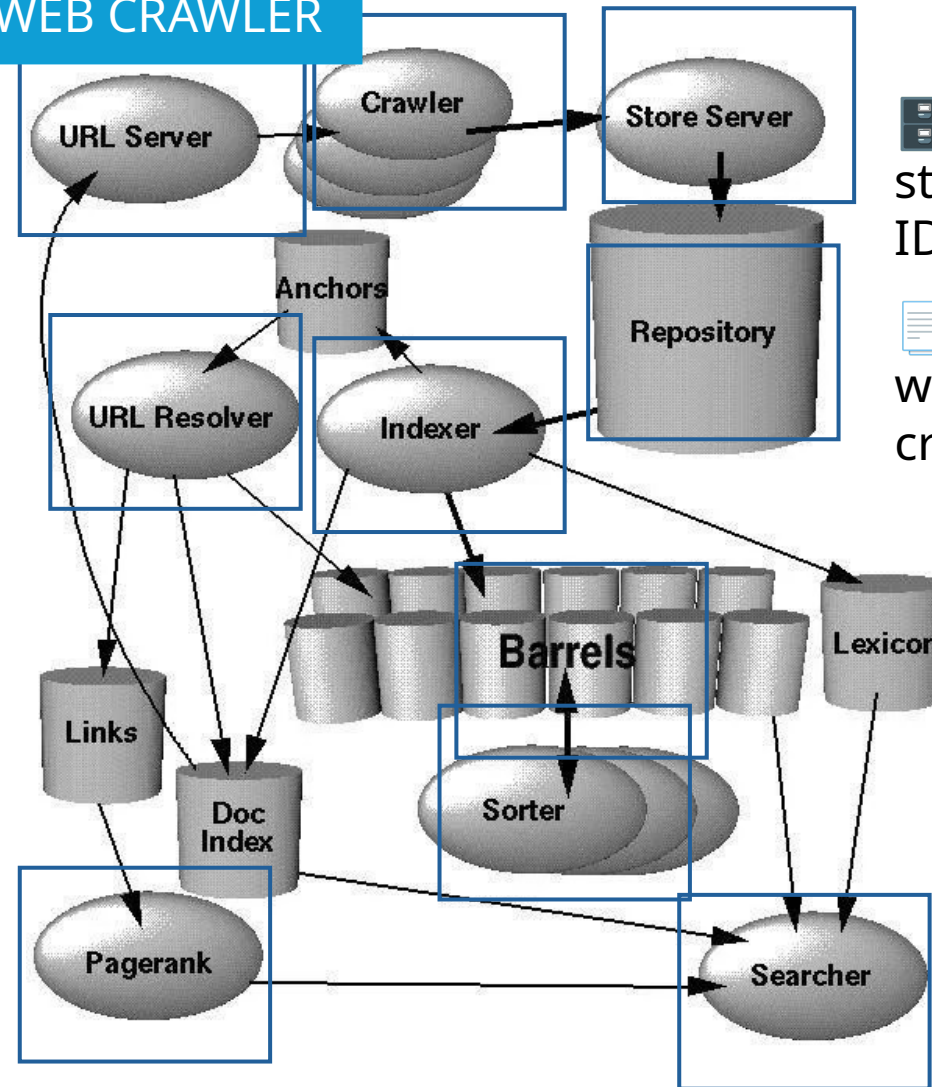
 **Store Server** : Compresses, stores pages & assigns document IDs.

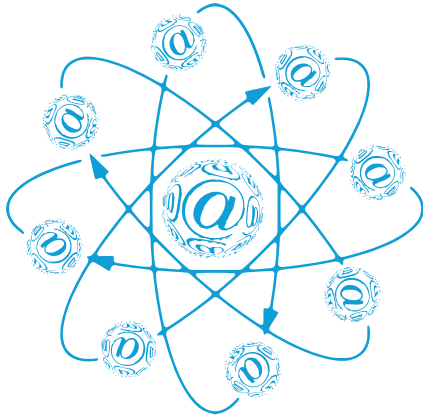
 Stores the raw, compressed web pages collected by the crawler

•  Serve as intermediate “buckets” for Hits (term occurrences) grouped by term-ID range

 **Sorter**: Generates inverted list from barrels and restashes them.

 **Searcher**: Processes queries using PageRank, inverted lists & lexicon.





## Navigating the Web of Links: HITS (**Hyperlink-Induced Topic Search**) in Action



The web is vast and interconnected through links.



To navigate and rank these links, Google used the HITS algorithm.



HITS helped identify authoritative pages with many inbound links.



HITS also found hubs, or pages linking to many authorities.



HITS boosted relevance by iterating between hub and authority scores.



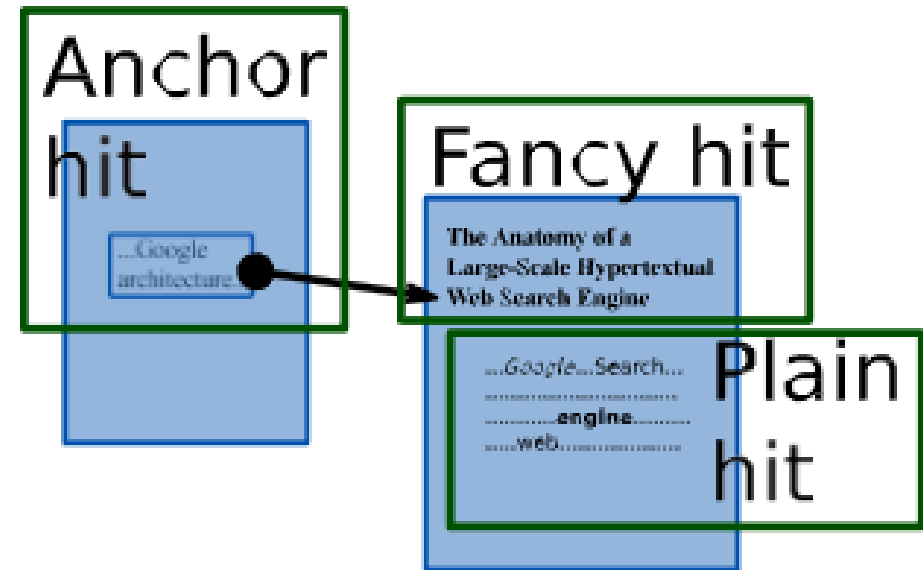
Google eventually transitioned to PageRank, but HITS laid the foundation for link analysis.



# Google Hit Types

Google categorizes term occurrences into three “hit” types for ranking relevance:

- ✨ **Fancy Hit**  
Occurs in prominent places like the URL, page title, or meta tags.
- 🔗 **Anchor Hit**  
A special fancy hit found in the anchor text of links pointing to the page.
- 📄 **Plain Hit**  
All other occurrences of the term within the body content.

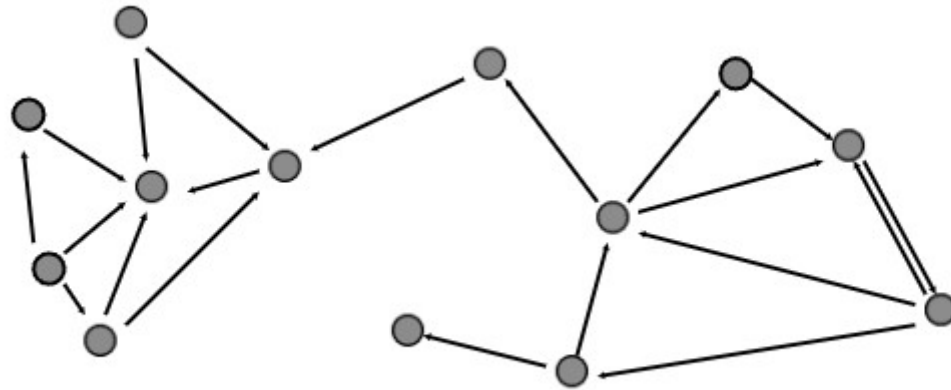


**But how to find the right link or HITS in the mess of links?**

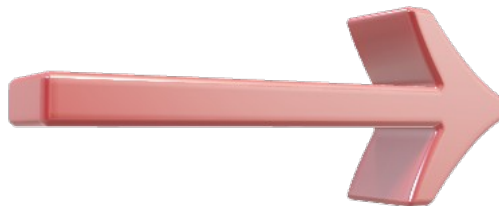
*Google's architecture efficiently calculates key signals for ranking documents based on queries using*

***RANKING FUNCTIONS***

*The Web is a cyclic graph of nodes (pages) connected by directed links (hypertext).*



Identifying important link using  
Link Analysis with [PageRank](#) and HITS



# PageRank

- PageRank measures a page's authority based on incoming links.
- The more (and higher quality) links a page receives, the higher its **authority**.
- PageRank values are computed iteratively, with each page passing its score to linked pages in proportion to the number of outbound links.
- The process continues until the PageRank scores reach equilibrium, adjusted by a damping factor to model user behavior



# Ranking Schemes in Web IR

The assumption behind Page Rank is that of a **random surfer**.

“Random surfer” clicks on successive links at random  $\rightsquigarrow$  equal probability of each link to be clicked

## Simple Ranking Scheme

- Step 1: Use the query (q) to filter documents (Similar to Boolean retrieval).
- Step 2: Rank the matching documents in order of their PageRank scores (PR(d)), from highest to lowest.

## Elaborated Ranking Scheme

- Combine multiple relevance signals for a more refined ranking.
- Example: Use a linear combination of:
  - RSV(d, q): The relevance score of document d to query q (e.g., using  $tf \times idf$  in the vector space model)
  - PR(d): The PageRank score of document d
- Formula:

$$RSV^*(d, q) = \alpha RSV(d, q) + (1 - \alpha) PR(d)$$

$\alpha$  is a parameter ( $0 \leq \alpha \leq 1$ ) that balances the influence of relevance and authority.

# Page Rank Calculation

## STEP 1

Iterative Computation of PageRank  $PR(p)$   
for a page  $p$

*Note : the Initial Page Rank for all pages  
is 1*

$$PR(p) = \overbrace{(1 - \beta) \times \frac{1}{N}}^{\text{Surfer gets bored}} + \underbrace{\beta \times \sum_{q \rightarrow p} \frac{PR(q)}{out(q)}}_{\text{Randomly following links}}$$

$N$ : Number of nodes / pages

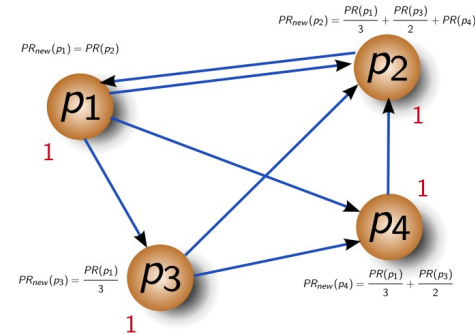
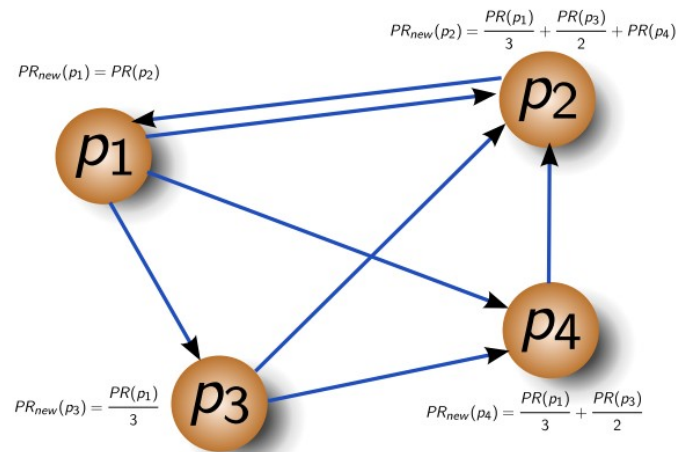
$\beta$ : Constant between 0 and 1

$out(q)$ : Number of outgoing links from page  $q$

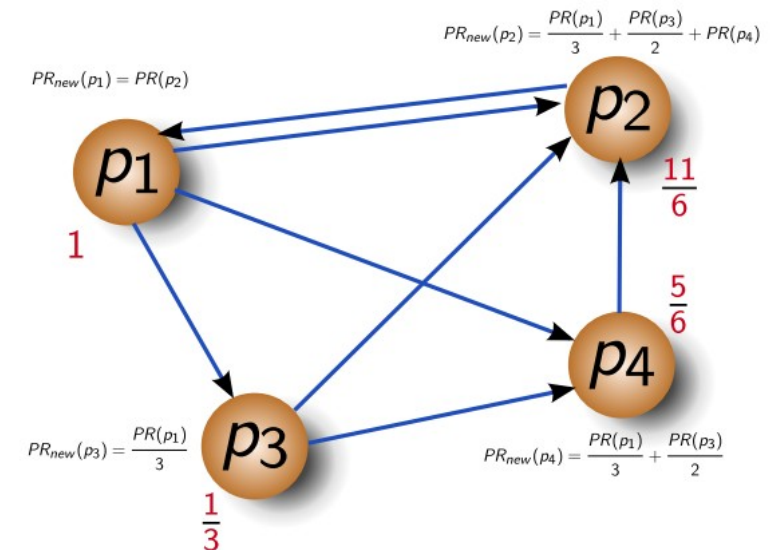
$q \rightarrow p$ : There is a link from  $q$  to  $p$

# Example Page Rank Computation

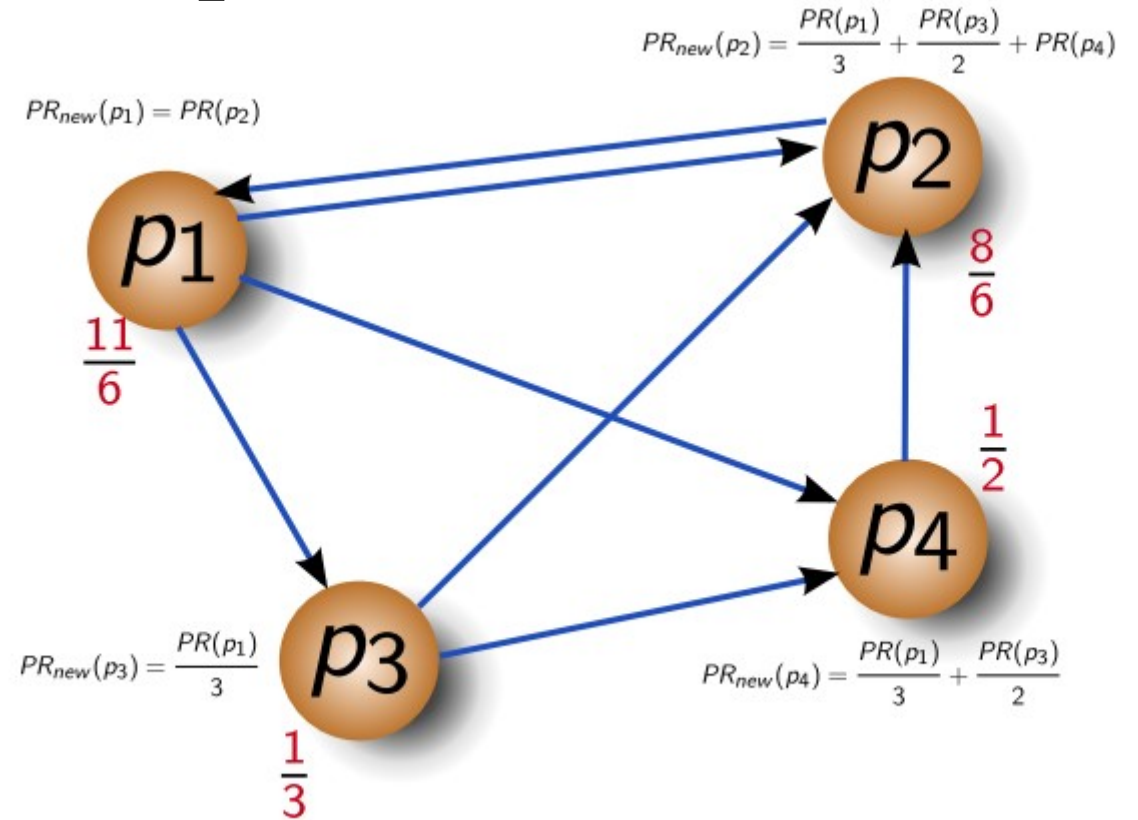
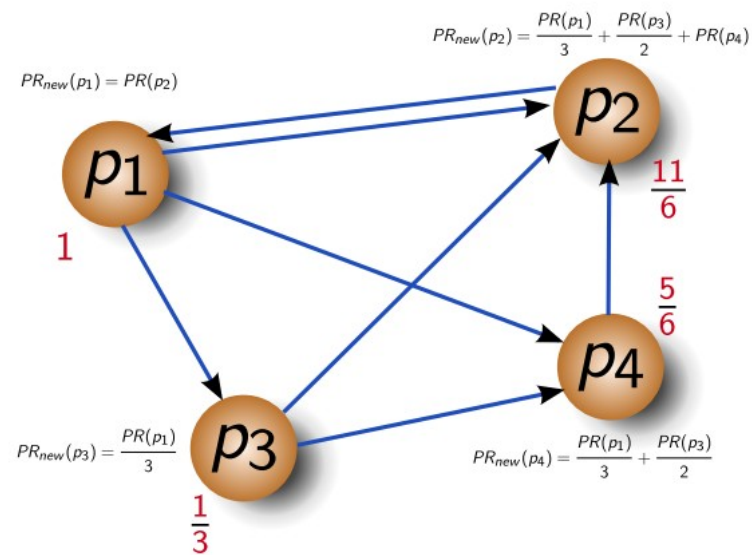
Example (set  $\beta = 1$ ):  $PR(p) = \sum_{q \rightarrow p} \frac{PR(q)}{out(q)}$



Note : Page Rank of all the links initially is set to be 1

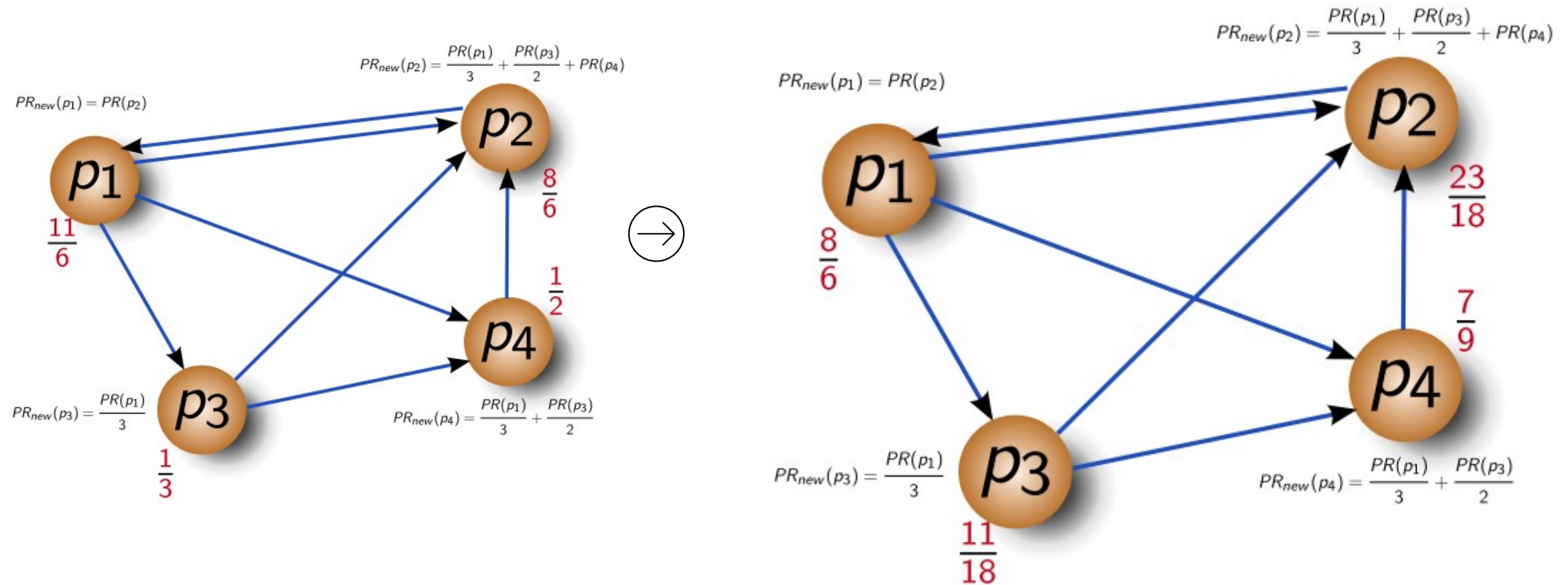


# Example Page Rank Computation

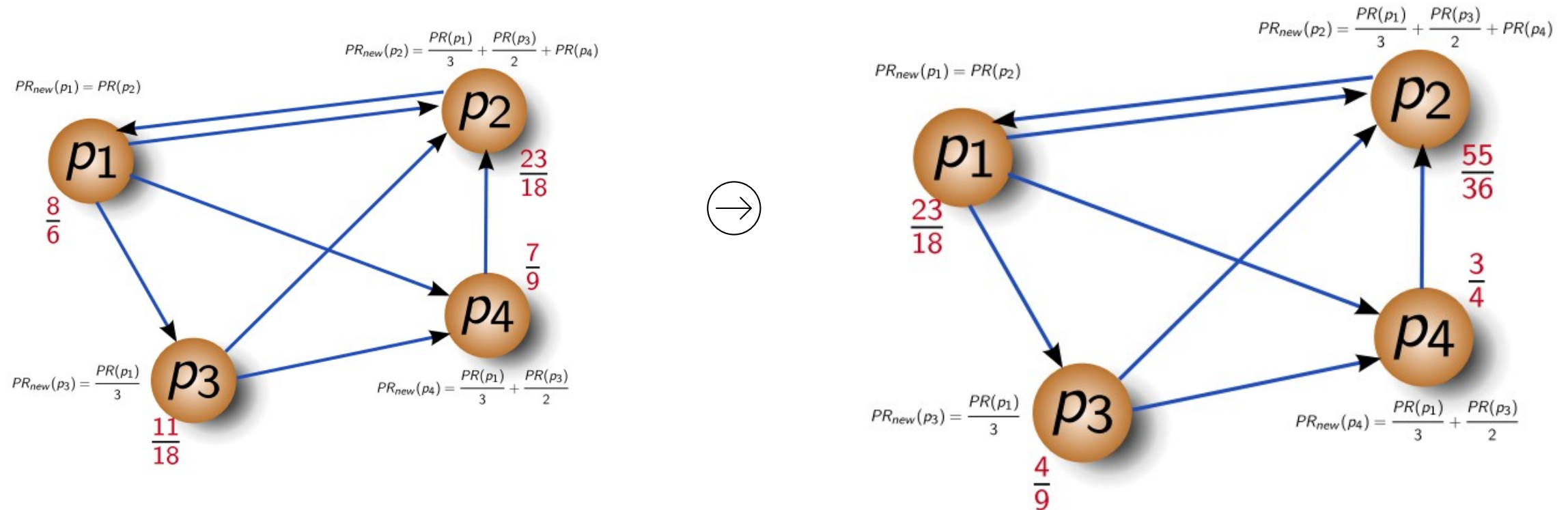




# Example Page Rank Computation



# Example Page Rank Computation



# PageRank: Strengths & Limitations

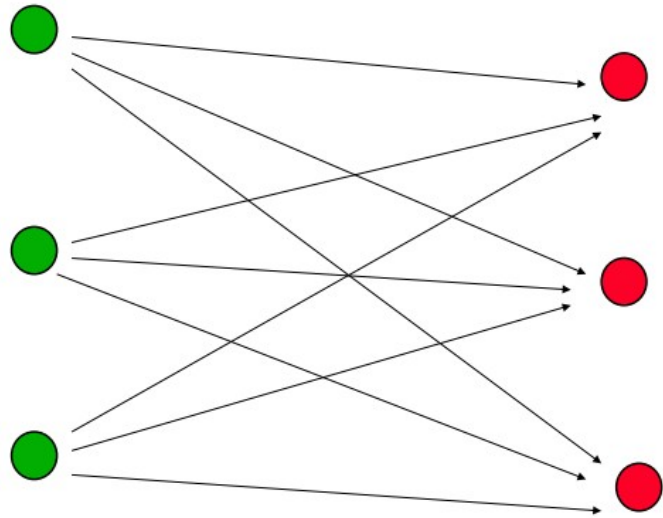
Page rank is Precomputed independently of queries — scalable to billions of pages

- **Strengths:**

- Highlights popular and authoritative pages
- Delivers strong results for homepage and general searches
- Favors entry points of websites

- **Limitations:**

- Less effective in niche or narrow domains
  - Vulnerable to manipulation (e.g., link farms)
  - PageRank can be boosted by buying irrelevant inbound links due to its content-agnostic nature
-



# Kleinberg's HITS Algorithm

- An algorithm for **ranking web pages** — just like PageRank, but in a different way.
- In HITS, **each page gets two scores**:
  - **Authority score**: Measures **how trustworthy and valuable** the page is (based on how many *good hubs* link to it).
  - **Hub score**: Measures **how good the page is at linking** to important pages (*good hubs* point to *good authorities*).

## 👉 Example:

- A university's main website could be a **hub** (because it links to many departments).
- A specific department page could be an **authority** (because it is important and many hubs link to it).

## In short:

- Good hubs link to good authorities.
- Good authorities are linked by good hubs.

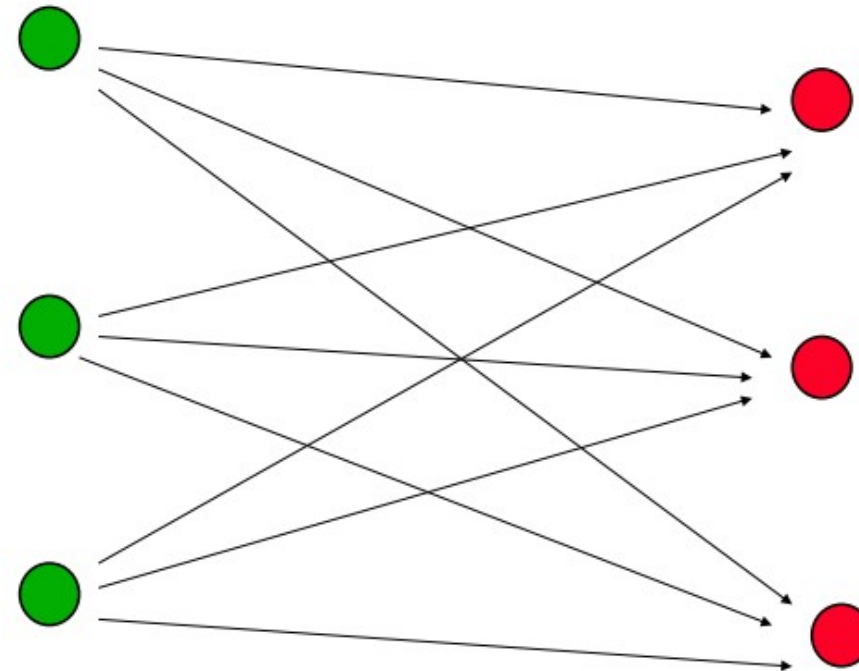
# HITS: 2-Step Algorithm

## 1. Subnet Selection:

1. Compute the RSV (Retrieval Status Value) for each webpage based on the query.
2. Select a neighborhood graph of relevant documents.

## 2. Score Computation:

1. Compute **hub** and **authority** scores for each page in the neighborhood graph.



# Hub and Authority Computation

Iterative Computation of authority  $a_p$  and hub value  $h_p$

$$a_p = \sum_{q \rightarrow p} h_q$$
$$h_q = \sum_{q \rightarrow p} a_p$$

$a_p$ : Authority weight for node/page  $p$

$h_q$ : Hub weight for node/page  $q$

and **normalisation criterium**

$$\sum_p (a_p)^2 = 1 \quad \text{and} \quad \sum_q (h_q)^2 = 1$$

---

# Hub and Authority Computation

## Outline of the Iterative HITS Algorithm

- **Initialize:** Set hub and authority scores of each page to 1.

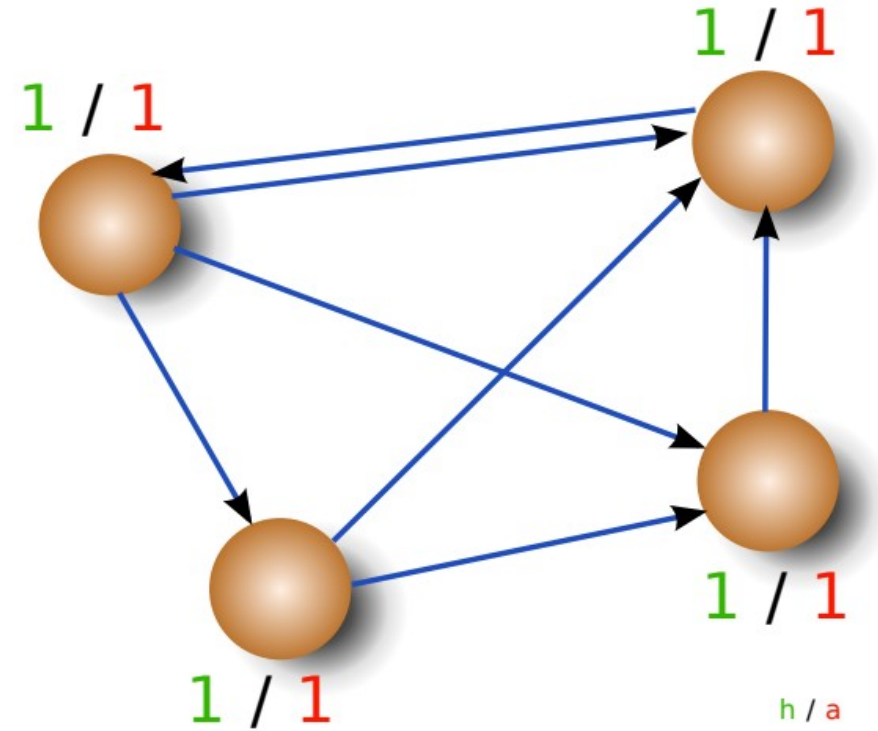
- **Update Scores:**

- Authority: Sum of hub scores of linking pages
- Hub: Sum of authority scores of linked pages

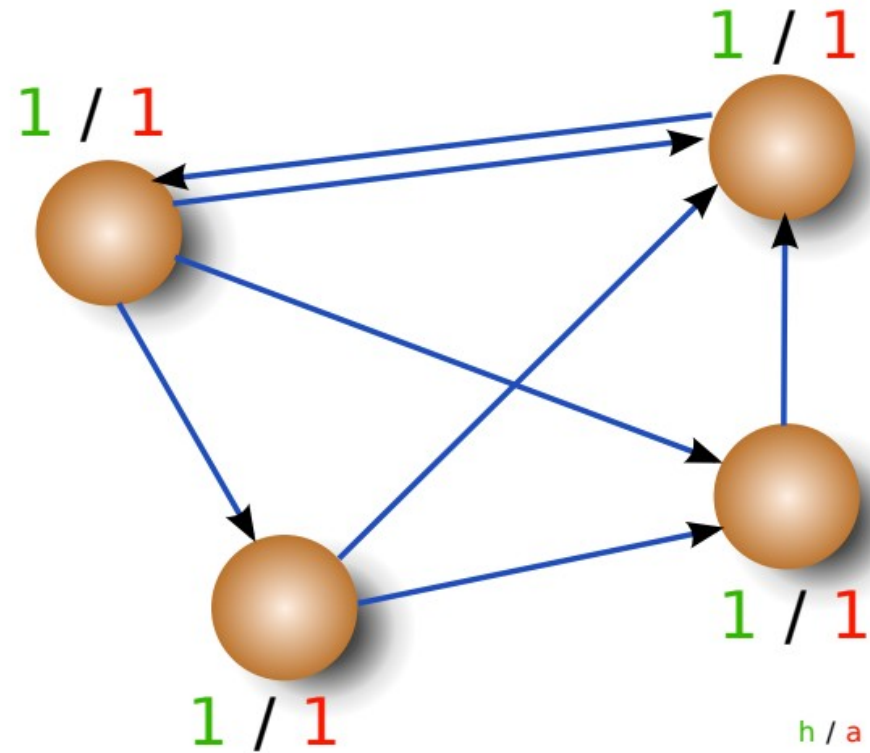
- **Normalize:**

- Divide each  $a_p$  by  $\sqrt{\sum_p a_p^2}$
- Divide each  $h_p$  by  $\sqrt{\sum_p h_p^2}$

- **Repeat** Step 2 until scores converge



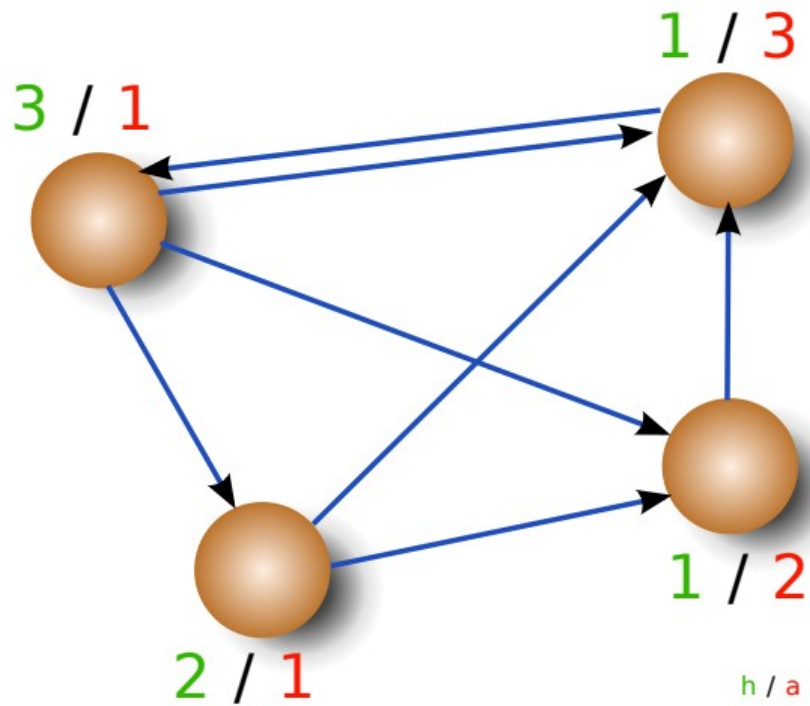
# HITS Example



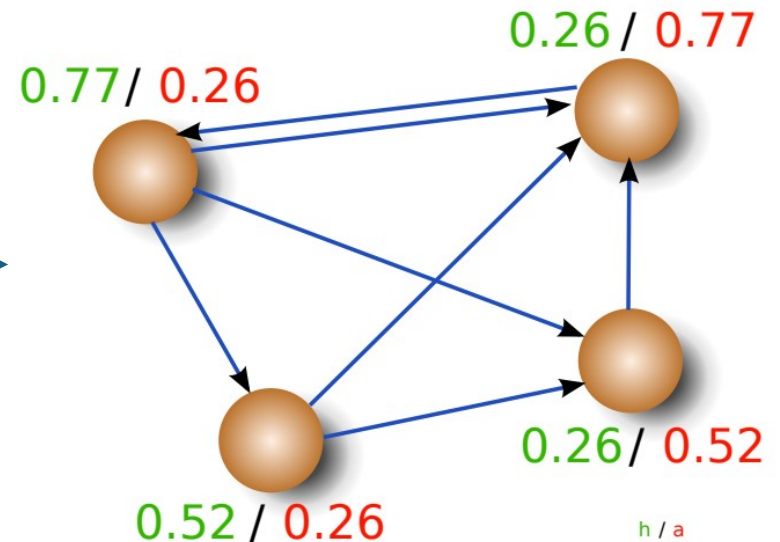
Let's start with a uniform **Hub** and **Authority** value for all the nodes



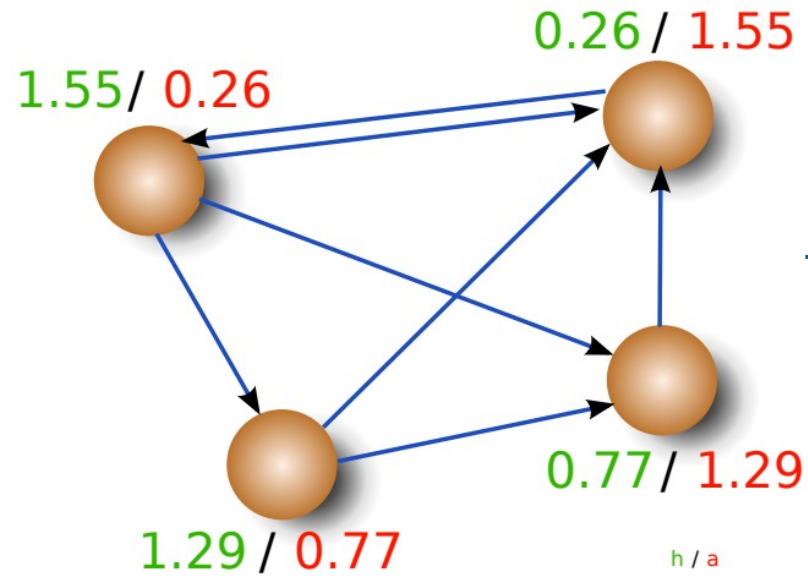
# HITS Example – Iteration I



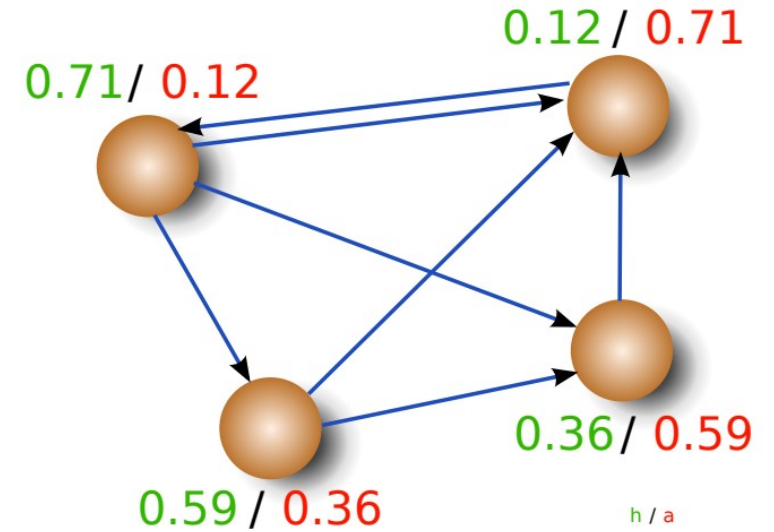
Normalisation



# HITS Example – Iteration II



Normalisation



# HITS: Strengths & Limitations








## ✓ Strengths

- Separates **hub** and **authority** values → enables richer search strategies
- **Authority scores** highlight important documents (similar to PageRank)
- **Hubs** act as quality surveys linking to authoritative sources
- Scores are **query-specific**, ensuring on-topic results

## ✗ Limitation

- Computation happens **at query time** → longer response times

# Topics Covered: Browsing & Searching the Web

-  The Web is a **graph** of interconnected pages
-  Search engines like **Google** use **RSV + Link Analysis**
-  **PageRank** identifies globally important pages
-  **HITS** separates **authorities** () and **hubs** () for topic-specific search
-  Link structure + content = **smarter, more relevant search**

 **Wishing you all the best for your upcoming exams & final assessments – go crush it !**

