PAGERANK

May 2023

1 Motivation

Information Retrieval is the term used for the process of searching for a information in response to a particular Query.Since Ancient Times,Information is stored in many forms like stones,walls,papyrus rolls,paper and now in the web. So it is Important that this Information is Retrieved Accurately when satisfying any Query.This leads to the Traditional Information Retrieval.

1.1 Trditional Information Retrieval

Traditional Information Retrieval is applied to smaller,non-linked collections of Information.This Information was mostly books,artworks,microfiche,CD and webpages. The Computerized Mechanism for searching items is known as Search Engines. Traditional Information Retrieval Method included Search Engines like **Boolean Search Engines,Vector Space Model Search Engines,Probabilistic Model Search Engines,etc**

1.1.1 Boolean Search Engine

It is based on the basic Boolean logic operators like AND,OR and NOT. It refers to a web page on the basis of a keyword.It judges on the whether a keyword is present or absent in a document.Thus it is a simple approach which is easy to implement.It is Fundamental model for many web search engines.But is not a efficient method because of many factors.One of the factors is Ambiguity.Ambiguity means a word or expressions having multiple meaning.This makes irrelevant pages getting shown up on a Query.Also the other factor is Synonymy,which refers to multiple words having similar meaning.This will create a problem if user uses a synonym of a word instead of the commonly used one. Thus this requires users to have a prior knowledge about the keyword they search.Hence it is not a efficient method though many web engines use boolean method in one way or other.

1.1.2 Vector Space Model Search Engine

Vector Space Model converts the documents and queries to vectors.This method overcomes the problems of ambiguity and synonymy.It works on the basis of the similarity between vector representation of query and document.Advanced Vector models returns the documents whose keywords are related semantically(having same meaning).But like Boolean method,Vector Space Model also has some limitations,since it stores documents and queries in form of vectors and matrix,it makes computation more complex and expensive.Thus it do not work well in large scale,it is limited only to small documents.

1.1.3 Probabilistic Models

Probabilistic Models works on the principle of probability of a document being relevant to the Query.It ranks pages on the order of the odds of the relevancy.It Operates Iteratively to obtain a Final Ranking probabilities.This requires a large amount of training data and are hard to program.Probabilistic Models assume that a term in document is independent of each other, ignoring term dependencies which make a search relevant.

Hence though there are different methods,none of them are fully effective which can make searching and information retrieval smooth.This changed completely with the introduction of World Wide Web,in 1989 which gave birth to Web Information Retrieval.

2 Web Information Retrieval

The Web is unique because web is very large,it is dynamic as it can be changed anytime,also Web is Self-Organized because anyone can post a webpage and link it anywhere and lastly one of the main feature of Web is that it is Hyperlinked. This linking system gave a more efficient and relevant searching systems and algorithms.

The Basic Elements of Web Information Retrieval Process are Crawlers,Page Repositories,Indexing Modules and Indexes.The main function of crawlers is to gather new information and webpage and store them in a repository until they are sent to indexing modules where a compressed form of the page is stored in indexes.

Query Modules is also an important element which converts a users language query into a language which the system can understand and compare with various indexes to seek information.This returns all the relevant pages to a query.

The Relevant Pages are then sent to a Ranking Module which ranks all the pages such that the ordered pages which show up at first are the most likely pages a user want to see.This is the most important Component of the Search

Engines because to select the most relevant page among many available related pages is very crucial.

3 Introduction

The Major Breakthrough in Ranking Module came in 1998,when Computer Science Doctoral Candidates Sergey Brin and Larry Page used Graphs and connected it to Search Engines.This led to the development of a very important algorithm-PageRank.PageRank is the algorithm developed by Larry Page and Sergey Brin,Founders of Google, in 1998 to rank webpages.PageRank has been used by Google to give a ranking system to webpages which helps during entertaining a Query in Google Search.

Pagerank algorithm works on the basis of computing a ranking for every web page based on the graph of the web page.This ranking is computed on the basis of links that points to it(incoming links).The pagerank score is the deciding factor of a web page in a web search.This score not only depends on the number of incoming links but also on the importance of the page that link to the given page.Hence a page which is linked by a higher rank score page,has more rank score than a page having more incoming links from low score pages.Further the pagerank also takes the random surfer model into consideration,which accounts for a user randomly clicking and surfing between web pages.This is quite important because practically a user will not travel linearly from page to page but sometimes can travel in a random manner.The random surfer model is applied by taking a damping factor.Pagerank score is calculated such that it is resistant to manupilative techniques such as spamming or artificial link exchange.

The Mathematical Formulation and derivation of the PageRank Algorithm was done by Larry Page and Sergey Brin, who are the founders of Google as a part of their research at Stanford University. Many concepts of mathematics are used in this formulation like markov chain theory , eigenvalues and eigenvectors etc. We will be seeing and studying those concepts and formulations in the upcoming sections.

4 The Initial Formula

Larry and Sergey began with a simple formula at first , which can be called as a 'simplified version of PageRank'. The formula was :

$$R(P_i) = \sum_{P_j \in B_{P_i}} \frac{R(P_j)}{|P_j|},$$

Here P_i is any Webpage , and $R(P_i)$ is the PageRank of that Page. B_{P_i} is the set of all the pages pointing to P_i Page or the set of all the pages which have

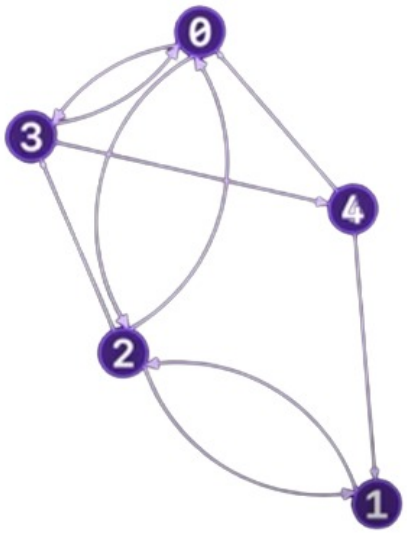

Figure 1: Example-1

outlink going to the Page P_i.$|P_j|$ refers to the total number of outlinks from that page. So, according to this formula we can say that the PageRank of a page is the sum of PageRanks of all the Pages Pointing to a page divided by the he total number of outlinks from that page.

In the above Figure 1, we can see a example of how the webpages/nodes are interlinked with each other. But initially we don't know the value of $R(P_j)$, i.e for calculating $R(P_0)$ we need to know the value of $R(P_2)$,$R(P_3)$ and $R(P_4)$.

To counter this problem, we can use an iterative procedure for this formula. Initially , we can assume that all the pages have same PageRank , i.e , $R_0(P_i)$=1/n, where n is the total number of pages(or nodes).
So, the formula can be rewritten as ,

$$R_{k+1}(P_i) = \sum_{P_j \in B_{P_i}} \frac{R_k(P_j)}{|P_j|},$$

where k represents the current iteration and we will successively perform the iterations for this equation until the PageRank scores converge to a stable value. This stable value will be the final Pagerank Score of the page.

4.1 Algorithm in the form of Matrix

Instead of representing in summation form we can represent it in matrix form like:

$$\pi^{(k+1)T} = \pi^{(k)T} H$$

here π^k is the PageRank Vector which contains PageRank Scores of All the Pages(Nodes) at k^{th} iteration.

$$\pi^0 = \begin{pmatrix} 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \end{pmatrix}$$

This represents the PageRank Vector of above example initially,where all the scores will be same.

The Matrix H for this Example is:

$$\mathbf{H} = \begin{pmatrix} 0 & 0 & 1/2 & 1/2 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1/3 & 1/3 & 0 & 1/3 & 0 \\ 1/2 & 0 & 0 & 0 & 1/2 \\ 1/2 & 1/2 & 0 & 0 & 0 \end{pmatrix}$$

Here H matrix is a Sparse matrix(Most of its elements are Zero) and thus they require minimal storage space to store its non-zero elements.Also H matrix is very similar to probability matrix of Markov Chains.

Markov Chains play an important role in PageRank because with the initial formula given by Page and Brin,problems like rank sinks,cycles etc took place.In our example if node 0 had no outer links then it will accumulate all the Pagescores with time and sometimes making all other scores 0,which is not ideal for a search engine.This is known as Rank Sinks.

Also,as seen in Figure 3,we have a graph where a loop is formed in which two nodes link to each other only.In this case the iterations will go on forever as it will never converge to a stable value instead it will flip-flop indefinitely forming a Cycle.To solve this we use Markov Chain Theory.

5 Markov Chain Theory

Page and Brin's PageRank can be said to be one of the applications of Markov Chain Theory although they have never mentioned Markov Chain Theory in

Figure 2:

their Papers.Then also we can say that Google's PageRank vector is a stationary Markov Chain.Some Important concepts of Markov Chain are:-

1)Stochastic Matrix

It is a non-negative matrix in which each row's sum is equal to 1.

2)Stochastic Process

It refers to a set consisting of certain Variables X_t.Here X_t represents the state of variable X at instant t(t is generally time).

A Markov chain is stochastic process in which a state of a Variable X at instant t+1 depends only on the State of the Variable at instant t.With Respect to Web,we can say that in the process of surfing through different webpages,the chance of a surfer visiting a webpage is dependant on the current page the surfer is in and not on the previous pages the surfer visited.Thus it is referred to as a random walk on the link structure.

3)Transition Probability

The Transition Probability p_{ij}(t) is the probability of being in a state S_i at instant t+1 given that chain is in state S_j at instant t.This is similar to our Pagerank Score $R_k(P_i)$

4)Transition Probability Matrix

It is a stochastic matrix for each value of t.Thus,H matrix is also a Transition Probability matrix.

5)Stationary Markov Chains

It consists of transition probabilities which does not change with t,i.e p_{ij} remains same at all instant.

5)Irreducible and Aperiodic Chains

It consists of transition probability matrix which is a irreducible matrix.Also all the irreducible chains which has primitive transition probability are known as Aperiodic Chains.

Now,with the help of Markov Chain Theory,we can convert the H matrix into a stochastic,irreducible and aperiodic matrix.By doing this the problems of Rank Sink and Cycles can be avoided.

6 Adjustments to the Model

Page and Brin used the Random Surfer notion to solve the problems in the initial formula.In this,a web surfer is randomly surfing through the Web.Arriving on a page with several outlink,the surfer chooses at random,any hyperlink to move to the next page.Thus,if a surfer has spends a large time on a particular page then the page must be important because the surfer while surfing randomly through the web,ended up returning to this particular page the most.

Again here the problem of dangling node(page with no outlinks) continues,this was solved by the Stochasticity adjustment.

1)Stochasticity adjustment

With stochasticity adjustment,any surfer if ended up in a dangling node,has a equal probability of starting again at any random page.Thus if we consider the graph of Figure-1,and consider there is no outlink from node-1,then the respective H matrix would be:-

$$\mathbf{H} = \begin{pmatrix} 0 & 0 & 1/2 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 1/3 & 0 \\ 1/2 & 0 & 0 & 0 & 1/2 \\ 1/2 & 1/2 & 0 & 0 & 0 \end{pmatrix}$$

but the respective **stochastic matrix** will be:-

$$\mathbf{S} = \begin{pmatrix} 0 & 0 & 1/2 & 1/2 & 0 \\ 1/5 & 1/5 & 1/5 & 1/5 & 1/5 \\ 1/3 & 1/3 & 0 & 1/3 & 0 \\ 1/2 & 0 & 0 & 0 & 1/2 \\ 1/2 & 1/2 & 0 & 0 & 0 \end{pmatrix}$$

Here, S is an updated H matrix with $\mathbf{S} = \mathbf{H} + \mathbf{a}(1/\mathrm{n}e^T)$, where $a_i = 1$ if page i is a dangling node and 0 if not.e^T is the vector of all 1.This made the resulting

matrix stochastic but it not enough for a stationary vector to converge to a unique solution.

2)Primitivity adjustment

With this adjustment it is considered that,a web surfer surfing through the hyperlink structure of Web many times may get bored and jump too a completely random page in the Web.Thus,the surfer might enter a new destination in URL link and surf to a random page in the web.To implement this Page and Brin,came up with a new matrix G such that:

$$G = \alpha S + (1 - \alpha)1/nee^T,$$

here α is **dangling factor**.It is a parameter of how much time the surfer follows the hyperlink structure,it is a scalar between 0 to 1.If α is 0.6,that means the surfer will follow the hyperlink structure 60% of the time and 40% of the time surfer teleports to a random page.Here the teleportation is random as $1/nee^T$ is equally distributed.

By making this adjustments we get G matrix which is stochastic,irreducible and aperiodic as well as primitive.Thus,it is clear that a unique positive π^T vector will exist and it will surely converge.Also we can represent G in the form of H matrix as:

$$G = \alpha H + (\alpha a + (1 - \alpha)e)1/ne^T.$$

Hence,the Final Formula of PageRank is:

$$\pi^{(k+1)T} = \pi^{(k)T} G.$$

7 Computation Of PageRank Vector

The PageRank Vector can be stated in two ways which are:-

1).Solving for π^T considering G as an EigenVector.

$$\pi^T = \pi^T G$$

Here,Primary focus is to find the Eigenvector of G with the corresponding eigenvalue λ_1=1(Since G is stochastic matrix).After getting the PageRank vector we normalize it so that its sum remains 1.The π^T is a stationary vector of markov chain.Considering the fact π^T is a probability vector.Hence,

$$\pi^T e = 1.$$

2).Solving for π^T in the linear Homogeneous System:

$$\pi^T (I - G) = 0^T,$$

In this system,the goal is to find the vector I-G,which is normalized to sum of 1.Also,again π^T is a probability vector.

$$\pi^T e = 1.$$

Though there are many ways to find π^T vector,Larry and Brin used one of the oldest yet simple method of iteration to find the stationary vector of Markov chain.They preferred this method because of many reasons,one of the main reason is its simplicity.The power method is applied to extremely sparse H matrix.Also the vector-matrix complexity is O(n) for this.The other reason is that power method only requires the matrix during vector-matrix multiplication and no matrix manupilation is done.Also,the power method requires to store sparse matrix H,dangling node vector and current vector π^T,making it a storage-friendly vector.Upon repeated experiments and computation it was found that for power method only 50-100 iterations are required before it converges to a stationary vector.This is very efficient because its complexity will be 50 O(n) as compared to other algorithms whose complexity touches cubic form.Also parallel Computation of product of rows can be done in this method.

Hence,keeping in mind several aspects,Power Method was selected by Page and Brin to compute the PageRank score.

8 Implementation of PageRank Algorithm to Other Field

- Airline route optimization means to optimize the route of airplanes in such a way that overall operational performance is enhanced resulting in maximizing efficiency and improving overall passenger experience.
- The pagerank algorithm used by google to rank web pages can also be implemented over here.
- Using the pagerank algorithm as the basis to explain the airline route optimization. Following are some basic analogies:

8.1 Defining the Network

- There are two components in the Graph theory: one is node and the other is edge .Here,we are considering airports as nodes and the path connecting two airports (i.e.Flight Routes) as edges .
- We will assign an initial rank to every airport (node). The initial rank is based on various factors such as connectivity, airport size, passenger demand or historical data etc.Here , the rank signifies the importance of a particular airport in the network.

- For example let us take an instance of a simple airline route network of three airports. Let the airports (nodes) be named as A,B,C. The flight routes are as follows

  ——- airport A $\rightarrow$ B
  ——- airport B $\rightarrow$ C
  ——- airport C $\rightarrow$ A

- A,B and C are nodes and AB, BC ,CA are edges

8.2 Damping Factor

- As we know that in the Pagerank algorithm used by Google if the markov chain is aperiodic and non-irreducible , there is a possibility that the user will fall into a webpage which has no outgoing links (i.e. a trap state). Now to overcome this anomaly, we introduce the ' damping factor(d)'.

- When you are stuck in a trap state (i.e state with no outgoing links) the damping factor gives the probability to jump to a webpage with high probability instead of jumping to a random webpage.

Similarly, when you are stuck in an airport and there are no direct flights to your target airport, then damping factor will give you another airport in between your target route which is the most feasible (i.e webpage with highest probability) and then you will proceed to your target airport via the airport given by damping factor.So the damping factor here is basically the probability of a passenger to choose a connecting flight which is more convenient to him (i.e airport with highest probability) to his target location when there are no direct flights.

What will be the value of the damping factor ?

- In the original PageRank algorithm developed by Larry Page and Sergey Brin at Google. The value of the damping factor d is set to '0.85 '.This value is commonly used as it is shown to provide optimum results in practice.The remaining 0.15 is then distributed among the nodes uniformly in the graph.r route via C or D

Figure1(a) shows an aperiodic and non-irreducible Markov Chain:

- So here are four airports A,B,C,D. Suppose you are at node A and want to reach node B so there is no direct route connecting A and B you have to take another route via C or D

- Let us consider the probability of AD route is 0.5 and that of AC route be 0.4 we will multiply the damping factor 0.85 with these two and whichever value will come will be the probability of passenger choosing that route whichever value comes the highest is the most feasible route.

- So in this example route AD is the most feasible route so the passenger will reach B via D.

8.3 CALCULATING THE RANK

- The repetitive PageRank algorithm is used to calculate the rank of each of the Airport.The process involves allocating initial ranks to each airport and then iteratively updating their ranks based on the number of airports linked with it. The iterative calculation process is essential for updating the rank values of each airport node based in weighted incoming edges (i.e inlinks to the node)
- Here, the incoming edges or the inlinks are the number of flights directed to a single airport. For example , each airport in the above graph has exactly 1 incoming link.
- This process continues until the values of the rank values converge.
- you can also fix the number of iterations so that stopping criteria is met. so that algorithm will perform iterations for a fixed number of times ,regardless of the Pagerank values .
- The following graph of rank vs iteration shows the convergence of ranks after a lot of iterations

8.4 RANK BASED ROUTE SELECTION

- Once the airport rank values converge , we can rank airports based on their significance in the route network.
- Highly ranked airports are likely to have sophisticated connectivity , good airplane management infrastructure , better facilities and profitable passenger demand.
- We will use these ranks to direct route selection and prioritize the routes between airports of higher rank values.
- The route optimization process is iterative after all pagerank calculation is done and the routes are selected we can recompute the rank of the selected routes. This process is known as iterative refinement and is used to optimize the network and improve overall efficiency of the airline routes.

8.5 Implementing the Mathematics:

Step 1 : Defining the nodes

- Let's take an example consider a 5 airport network .Let the airports be Delhi(D) , Mumbai(M) , Bangalore(B) , Chennai(C) and Kolkata(K)

Step 2 : Constructing The Transition Matrix

- The transition matrix contains the probabilities of transitioning from one node to another. Let us assume the following probabilities based on passenger demand and other relevant factors

  The Matrix H for this Example is:

$$\mathbf{H} = \begin{pmatrix} & \mathbf{D} & \mathbf{M} & \mathbf{B} & \mathbf{C} & \mathbf{K} \\ \mathbf{D} & 0.2 & 0.4 & 0.2 & 0.1 & 0.1 \\ \mathbf{M} & 0.3 & 0.2 & 0.1 & 0.2 & 0.2 \\ \mathbf{B} & 0.1 & 0.2 & 0.3 & 0.1 & 0.3 \\ \mathbf{C} & 0.2 & 0.1 & 0.1 & 0.3 & 0.3 \\ \mathbf{K} & 0.4 & 0.1 & 0.2 & 0.1 & 0.2 \end{pmatrix}$$

- In this matrix the rows represent the current state and columns represent the next state. The values in the matrix represent the probabilities of transition from current state to next state (airport) .For example the probability of transitioning from Chennai (C) to Kolkata (K) is 0.3.

Step 3 : Assigning initial values

- Initially , we will assume that all the airports have uniform probabilities . Since there are 5 airports , we provide a probability of 0.2 to each airport:
- Initial probability vector : [0.2 , 0.2 , 0.2 , 0.2 , 0.2]
- What this actually implies is that each airport has equal probability of becoming the starting airport of a passenger.

Step 4 : Looping the Markov Chain

In this step, we will perform iterative calculations using the transition matrix to update the probabilities of being in each state (airport). This involves taking the product of the current probability vector with the transition matrix to obtain the next probability vector. We will repeat this process until convergence is achieved or for a predetermined number of iterations.

Calculation:

- Initial Probability Vector: [0.2, 0.2, 0.2, 0.2, 0.2]
- **Iteration 1:** [0.2, 0.2, 0.2, 0.2, 0.2] x Transition Matrix
- **Updated Probability Vector (Iteration 1): [0.22, 0.24, 0.18, 0.18, 0.18]**

- **Iteration 2:**[Updated Probability Vector] x Transition Matrix
- **Updated Probability Vector (Iteration 2): [0.222, 0.232, 0.194, 0.19, 0.162]**
- **Updated Probability Vector (Iteration 3): [0.2234, 0.2276, 0.1956, 0.1908, 0.1626]**
- **Updated Probability Vector (Iteration 10): [0.2248, 0.2269, 0.1956, 0.1932, 0.1595]**
- **Updated Probability Vector (Iteration 12): [0.2243, 0.2253, 0.1955, 0.1947, 0.1602]**
- **Updated Probability Vector (Iteration 50): [0.2245, 0.225, 0.1955, 0.1949, 0.1601]**
- **Updated Probability Vector (Iteration 100): [0.2245, 0.225, 0.1955, 0.1949, 0.1601]**
  .....
- **Iteration n:** [Updated Probability Vector] x Transition Matrix

This process is repeated either for a number of iterations or until the convergence of ranks is achieved.

In this case , after performing 100 calculations , the changes in the probability vector are insignificant . We can say that the obtained probability vector is our steady state distribution for the markov chain

Step 5 : Analyzing the steady state distribution

- The steady state distribution that we obtained represents the long-term probabilities of being in each airport.These probabilities indicate the relative significance of the airports within the network.
- Steady State Distribution : **[0.2245, 0.225, 0.1955, 0.1949, 0.1601]**
- Analyzing the steady state distribution, we can observe that Mumbai (M) has the highest probability of approximately 0.225, followed by Delhi (D) with a probability of 0.2245. This suggests that Mumbai and Delhi are relatively more important or frequently visited airports in the network. Bengaluru (B), Chennai (C), and Kolkata (K) have lower probabilities, indicating their lesser significance in terms of air traffic or connectivity .
- Based on this analysis, the steady state distribution can be utilized to optimize airline routes by allocating more flights or resources to airports with higher probabilities (such as Mumbai and Delhi) to maximize efficiency and meet passenger demands.

Step 6 : Optimizing Routes

- Utilizing the above result to Optimize Airline Routes. Allocating more resources or flights with higher probabilities indicating their importance and influence over other airports in the network.this optimization can help in increasing the efficiency of the airline route network. This will also help in free flow of passengers.
- **[0.2245, 0.225, 0.1955, 0.1949 , 0.1601] [Delhi(D), Mumbai(M) , Bangalore(B), Chennai(C), Kolkata(K)]**

  Based on these probabilities, we can recognize Mumbai(M) as the airport with the highest probability (0.225) after n iterations designating its significance, importance and influence in the country.
- To optimize Routes, we can allocate more flights and resources to these high Probability airports . Here it's Delhi and Mumbai.

How can we achieve that ?

- **Increase flight frequency :** Increase the number of flights between Delhi and Mumbai so that we can enhance connectivity and provide passengers more options ,free flow and convenient travel between these two cities.
- **Upgrading Facilities :** Investing in infrastructure at both the airports in order to handle the increased passenger traffic and providing a seamless traveling experience.This may involve expanding(adding) the terminal airport (making it huge), streamlining security , luggage handling process, making it automated at basics etc
- **Preferred Scheduling :** Optimizing flight schedules between Delhi and Mumbai based on passenger preferences and demand patterns.Offer flights at various times a day to accommodate different travel needs and improve passenger convenience.
- **Expand Route Options :** Introducing additional routes connecting Delhi and Mumbai to other high demanding destinations.(which has a comparatively high probability next to them and has got a high importance score).