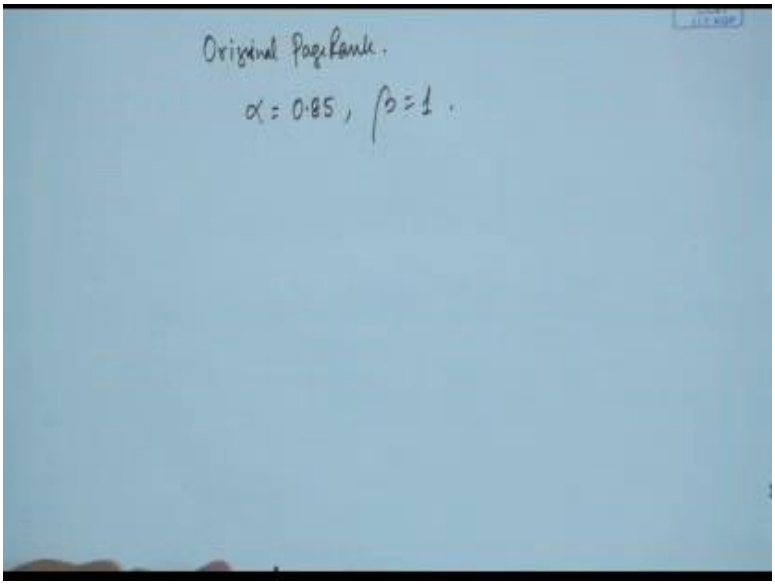
Complex Network: Theory and Application
Prof. Animesh Mukherji
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 06
Network Analysis – V

Welcome back. Last day we have talked about the idea of calculation of Page Rank, and we have also computed explicitly the formula that goes into calculating the value of Page Rank.

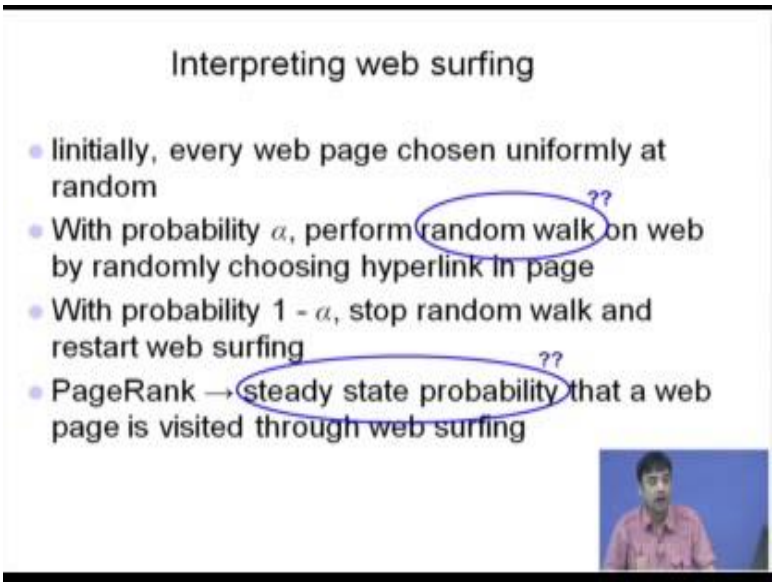
(Refer Slide Time: 00:39)



In the original Page Rank algorithm alpha was set to 0.85 and beta was set to 1. It is not known why is this parameters where chosen, but possibly the best ranking results in terms of user satisfaction was probably obtained if the constants where set like this. Now, today we will try to dig a little bit deeper in to this idea of Page Rank. The question is like what background process like related to the user actually can result into this formulation of Page Rank. That is the question that we are going to ask.

So, there is an underline process that actually really mimics the formula of Page Rank that is the question that we are going to ask and trying to see if there is a satisfactory answer to this question. This brings us to what we call the random surfer model or interpretation of the web surfing as a random process.

(Refer Slide Time: 02:00)



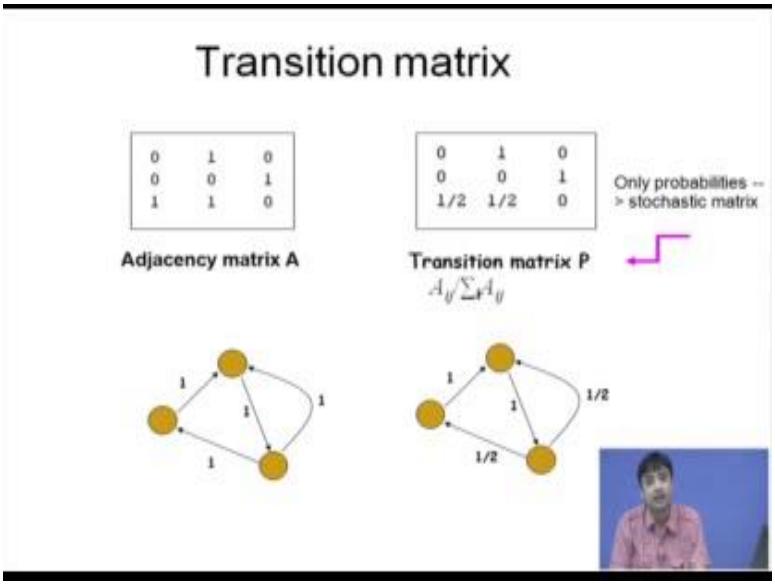
So, if you look at the slides the steps of the algorithm are very simple. Initially, every web page is chosen uniformly at random. Suppose you are on the world World Wide Web graph and you choose every individual webpage that is every individual node with equal probability that uniformly at random we choose one of them. Now again look at this slide, so with probability alpha perform random walk on web by randomly choosing hyperlink in the page with probability $1 - \alpha$ stop the random work and restart web surfing. Page Rank is nothing but the steady state probability that a webpage is visited through the web surfing. The idea is very, very simple.

So, what you do here you are standing on a node on the network, now from that node you move to any one of the neighbors of that node uniformly at random that is what is we refer to as random walk. Now this you do with some probability say alpha. And with the other probability $1 - \alpha$ you stay back in that particular node and you read the contents of that particular page, that is the probability $1 - \alpha$.

So now, if you continue doing this process for a quite long time then the stationary probability that you will land up on a particular node is what is the Page Rank value. That is what actually corresponds to the value of the Page Rank. As I said there are two important things to understand; one is the random walk which I have already try to explain you we will see example to make things better, and there is this steady state

probability. These are the two important keywords that we should get to know while understanding random web surfing.

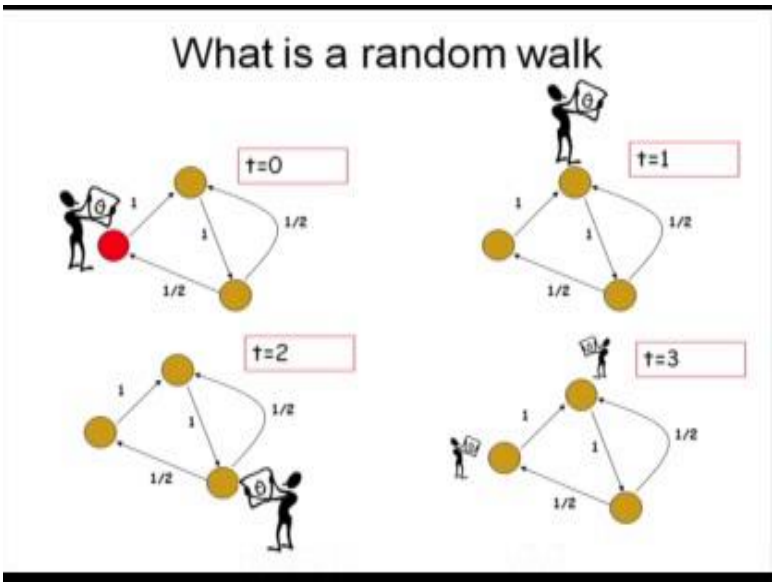
(Refer Slide Time: 03:51)



So, the first of this is the idea of random walk. Again look at this slide. Let us say you have an adjacency matrix A of this form. So, from this adjacency matrix you convert it to a probability matrix a stochastic matrix, where you normalize the contents of each row by the sum of that row. That is you are basically normalizing the each cell by the degree of that node. You normalize the matrix and you get a transition matrix, this is what is called the probability transition matrix or the row stochastic matrix. So, that is basically A_{ij} the content of each cell by $\Sigma_i A_{ij}$

Now given this structure, suppose you have the original graph the graph corresponding to the adjacency matrix A in this particular case would look like the one that I am pointing by the mouse. Now when you translate this graph into a transition matrix the values on the edge correspondingly change. As you see in the transition graph. Now, given this probability matrix are the transition matrix, this stochastic row, stochastic matrix are the transition matrix. The task that you have to do is the following.

(Refer Slide Time: 05:20)

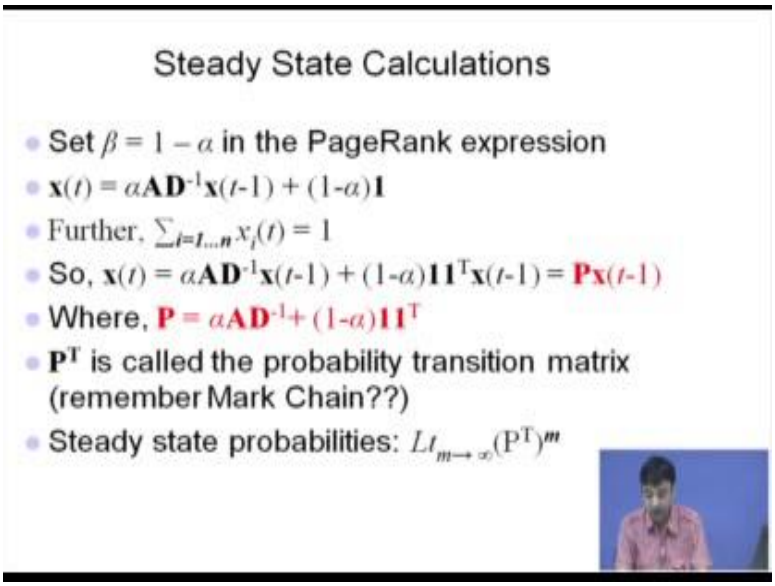


Suppose, there is a random walker standing at the node red node A here; now, for this walker there is only one place to go and that is this particular node at this time point t = 0, so it moves there. Now from here, there is again only one possibility and that possibility is to come to this particular node here. So, the random walker moves from this particular node to this particular node. That happens at time instant tool.

Now from here the random walker has equal chances, see there is a probability half of going to this node and there is a probability half of going to this node back. So, the random walker can take any one of these two paths. Probably it can go here either here or you can go here, each one with probability half.

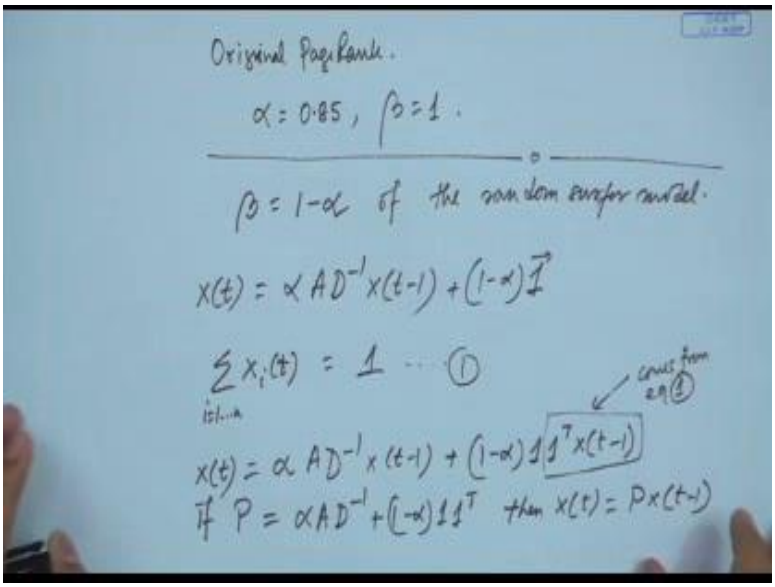
So, in this way you can try to simulate this process. This is the process that we are talking about, this is the process of web surfing. So, every time the walker with some probability alpha jumps into one of the other web pages with 1 minus alpha stays back in that particular webpage. So, that is kind of simulated in this slide.

(Refer Slide Time: 06:40)



So, now given this situation we can always write the Page Rank equation back. Now, the probability that the walker stays on the page that is your inherent popularity, this would correspond to my beta in the expression for Page Rank that we have already written.

(Refer Slide Time: 07:03)



Basically here, $\beta = 1 - \alpha$ probability of the random surfer model. So, my beta is the probability of staying in that particular node because that is my inherent centrality, and in this random surfer model that is set to $1 - \alpha$. The other part remains as alpha. Then you

can write X(t), the expression for X(t) that is the popularity value as $\alpha A D^{-1}X(t-1) + (1 - \alpha)1$ into the vector of all one's.

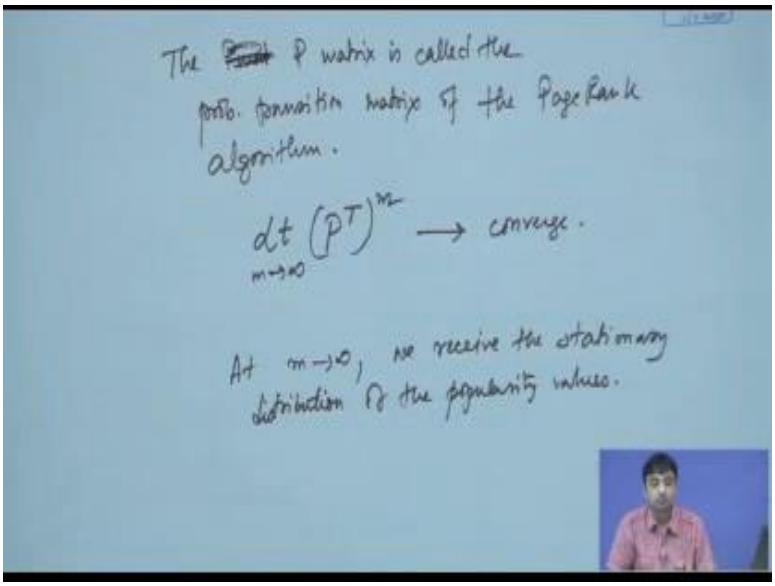
So, you can write the Page Rank expression once again. I have written the same Page Rank expression as we wrote earlier, but now substituting beta with $1 - \alpha$. Now this actually has an advantage we can further simplify the formula. If we assume that we are working on the stochastic matrix that is the transition matrix where everything is normalize between 0 and 1, so it is a stochastic matrix all the values are probability values. Then the sum of all X(t)'s, actually i 1 to n these values the sum of all the entries for all the X(t) values, so each node will have an X(t) value, so each node will have a popularity value and the sum of all these popularity is suitably rescale because we are working on the stochastic matrix. So that is why this sum will be equal to 1.

So given this, what we can write. We can rewrite the expression for X(t) as X(t) is equal to $\alpha A D^{-1}X(t-1)$, and we play a small trick out here $(1 - \alpha)1, 1^{T}X(t-1)$. Note that $1^{T}X(t-1)$ is nothing but 1, because you are multiplying a vector of one's with the entries of the X(t-1) vector. So, X(t-1)vectors will have the node popularity values for each of the individual nodes. If you multiply these values with the vector of all one's so it will be sum of all these values. Basically, it will be sum of all these X(t) values for each individual node. And this sum as we have seen is equal to 1; from the previous let us call this equation 1. This part actually comes from equation 1.

So, basically if we know that the sum of all the popularities values always remains between 0 and 1 in each step whatever we do its only the relative ordering of the x values changes, but the sum of the values always remains 1 then $1^{T}X(t-1)$ to be equal to 1, because you are making a product of a vector of all ones with the in individual x t values this will result into sum of all the X(t) values which is equal to 1.

So now, if we consider P is equal to $\alpha A D^{-1} + (1 - \alpha)1\ 1^{T}$, then we can write X(t) is nothing but P X(t-1). And this P matrix is called the Probability Transition Matrix of Page Rank.

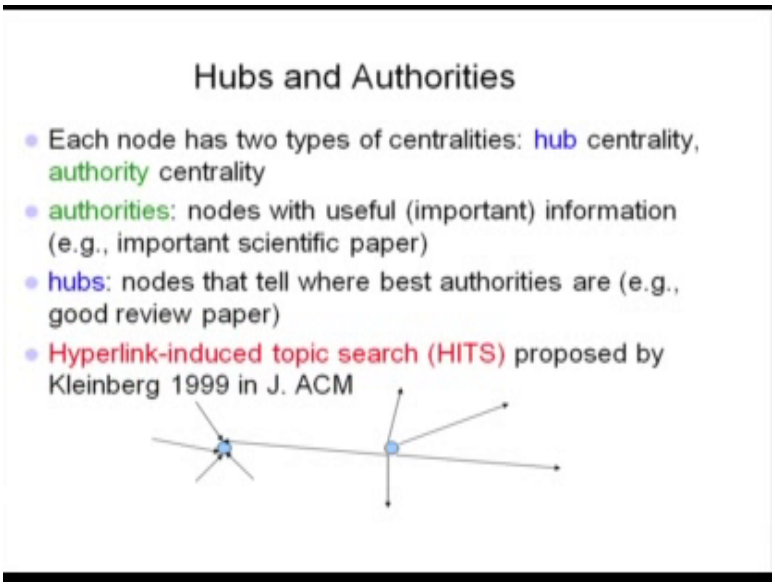
(Refer Slide Time: 11:15)



The P matrix is called the probability transition matrix of the Page Rank algorithm. And what we are interested in, we are interested in the limiting value of this matrix say for sum m limiting to tending to z infinity we say this should converge; $(P^T)^m$. Basically, what we are doing it in each step your powering this transition matrix and you are continuously doing this power, so as we are doing in the original eigenvector centrality case. So, you are continuously powering this value.

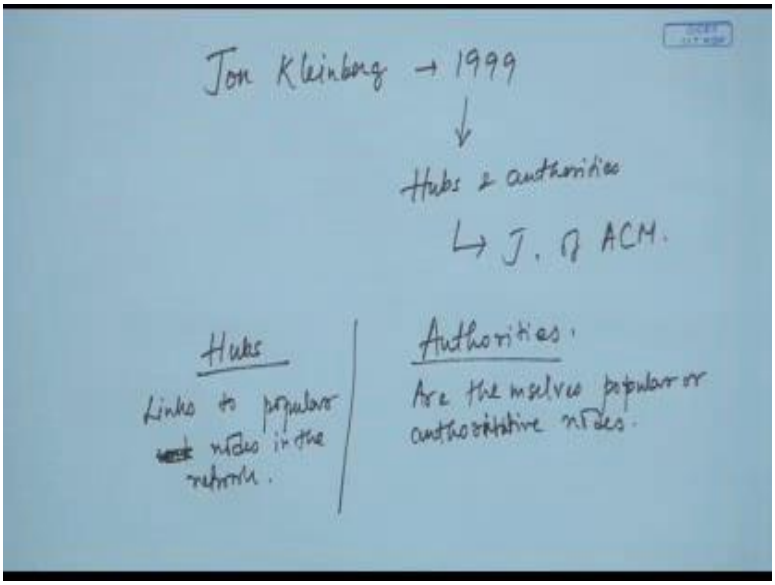
In the first step you have p in the next step, you have p^2 in the third step, you have p^3 and so on and so forth. And after a pointing time this transition matrix does not change any further and that is the point where you get the stable set of the stationary distribution of the popularity values. At large m, at m tends to infinity we receive the stationary distribution of the popularity values. That is how one can reinterpret the idea of Page Rank formula as a random process of web surfing.

(Refer Slide Time: 13:21)



So, once this idea of Page Rank came into business there were a lot of other people who started work in order to make this idea even better and better.

(Refer Slide Time: 13:48)



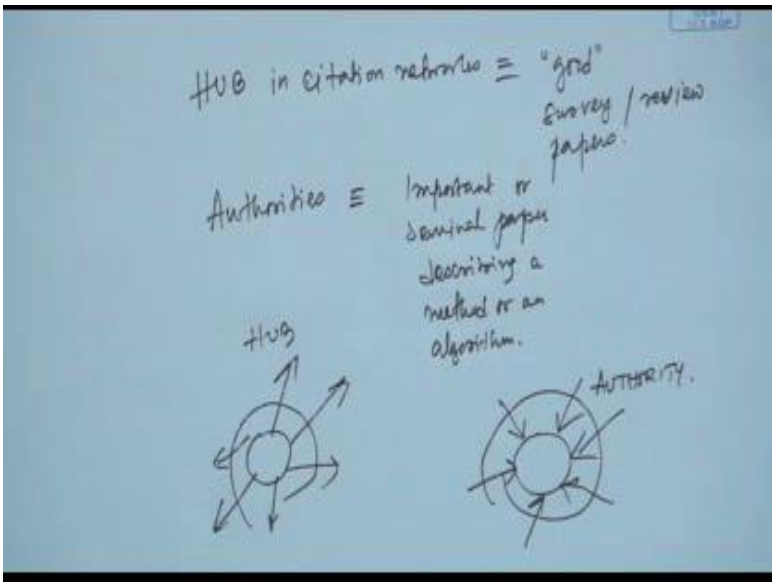
And one of the most notable works in this area was by our famous scientist called Jon Kleinberg, who in 1999 proposed the idea of hubs and authorities in the famous Journal of ACM. So, the idea again is very simple and it just an extension of the initial Page Rank idea, but it is actually found to be more impactful. The results that are obtained

using the hubs and authorities idea are found in general better than the Page Rank approach.

The idea is very simple. There are two different types of nodes. Basically like the Page Rank was having only one popularity value and that was based on the in degree of the nodes. If you remember correctly the node c that I showed you earlier in this slide was having a very high popularity value by virtue of having in degree from b. The Page Rank idea is mostly based on the concept of having high in degree values, having in degrees from highly popular nodes. Now Kleinberg extended this idea to also the out degrees and there comes the concept of hubs and authorities.

So, there are two types of entities hubs and authorities. The hubs, as the name suggests are links to popular nodes in the network, whereas the authorities are themselves popular or authoritative nodes themselves. This is easy to understand in the context of citation networks actually. So, basically in a citation network a hub node would be such a node which contains references or out degree or pointers to all important or authoritative papers. Such cases of hubs in citation networks would be analogous to survey or review papers.

(Refer Slide Time: 16:44)

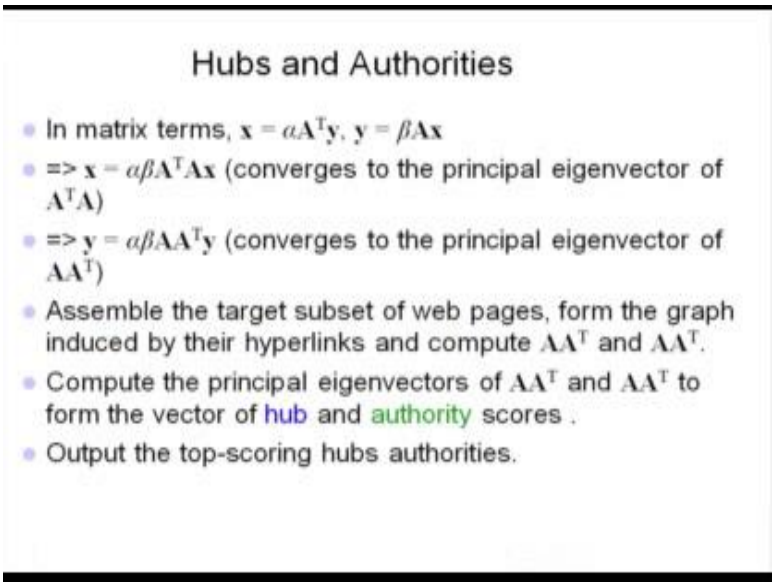


So, hubs in citation networks would be basically analogous to good survey or review papers, whereas authorities in citation network would be analogous to important or seminal papers describing a method or an algorithm. In other words the paper which

actually describes the shortest path algorithm by (Refer Time: 17:48) would be a authoritative paper, whereas in general survey papers on the subject of graph theory which actually refer to such papers like that of (Refer Time: 18:01) would be a example of a hub node in the context of the citation network.

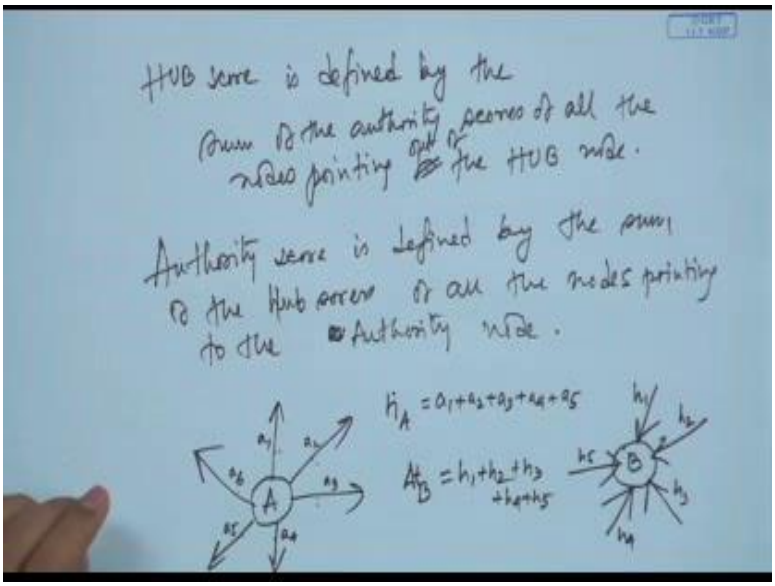
So, now the question is given this structure of the network how one should try to quantify the hubs and the authorities. Basically, from the definition itself it becomes clears that for a hub what is important is the out degree the out degrees, whereas for an authority what is more important is the in degree. Now, based on this in tuitions we have to develop a quantitative metric for both hubs and authorities.

(Refer Slide Time: 19:00)



So, the idea is very simple.

(Refer Slide Time: 19:09)

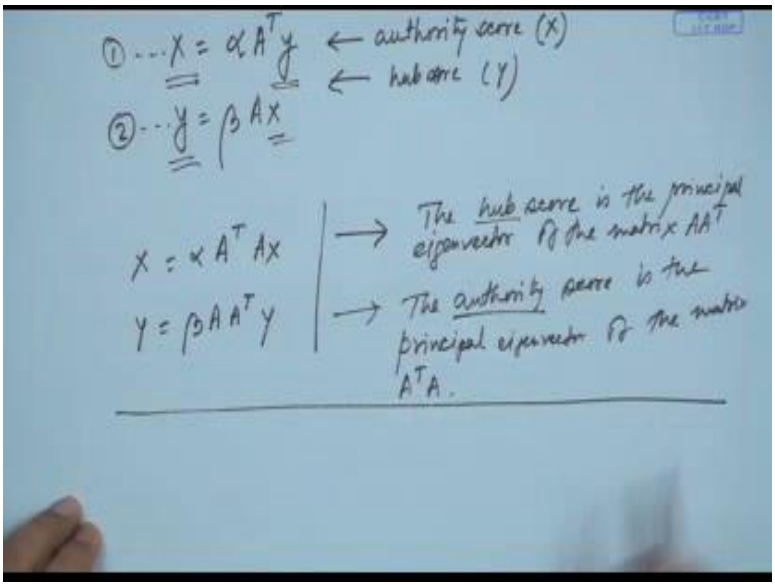


Now the hub score is defined by the sum of the authority scores of all the nodes pointing of the hub node; we will see an example out of the hub node. Authority score is defined by the sum of the hub scores of all the nodes pointing to the authority node. Suppose there is a hub node A here and it is pointing say to 5 other; 1, 2, 3, 4, 5, 6 other nodes. And suppose the authority scores of each of these nodes is defined as is given by say a_1 here, a_2, here, a_3 here, a_4 here, a_5 here, a_6 here. So, the hub score of the node A, if we denote it as h A should be equal to $a_1 + a_2 + a_3 + a_4 + a_5$

Similarly, the authority score can be computed. Suppose, there is this node B which is being pointed to by some hub nodes and say each of the hub scores of this hub nodes are h_1 , h_2 , h_3 , h_4, and h_5. Then the authority score of the node B, let us called a t of node B is expressed as the sum of the hub scores, h_4 and h_5. This is how you express the hub and the authority scores. And like Page Rank it is actually a recursive definition. So, your hub score is dependent on your authority nodes, they are neighbors, they are neighbors, and it continues it in this way.

Here, for simplicity I have shown a first level example, but this is actually a recursive definition just like that of the Page Rank or the initial eigenvector centrality idea.

(Refer Slide Time: 22:39)

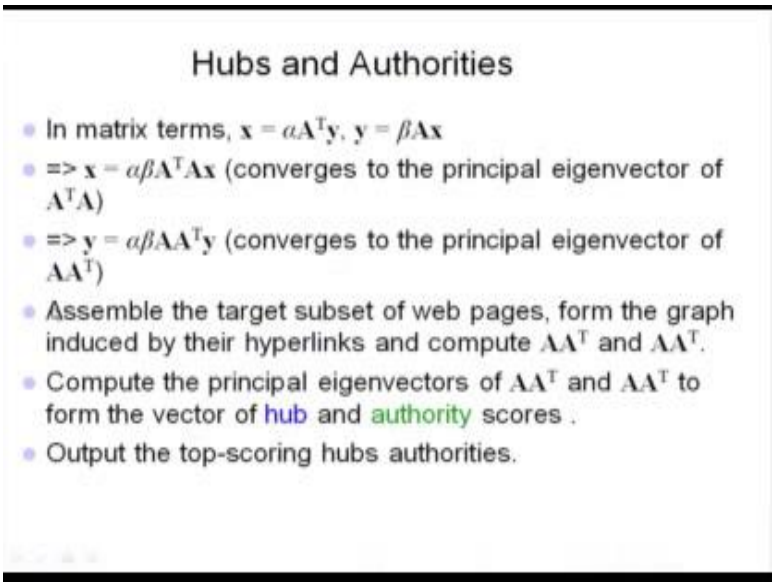


So then, how to express this in mathematically or quantitatively? So in matrix terms actually you can write the hub score as x equals $\alpha A^T y$, this is the authority score. So, basically authority score is given by x and hub score is given by y. And similarly you can write y is equal to $\beta A x$, just as we were doing in the previous examples for Page Rank and general eigenvector centrality.

So, the authority score is basically summed over all the hub scores, and the hub score is basically summed over all the authority scores. Given these two equations; let us call this equation 1 and this equation 2 here. Given this equations we can express x as $\alpha A^T A x$ and similarly we can express y as $\beta A A^T y$. That means, the hub score is the principal eigenvector of the matrix, so the hub score should scale as the principal eigenvector of the matrix $A A^T$ whereas, the authority score should scale as the principal eigenvector of the matrix $A A^T$

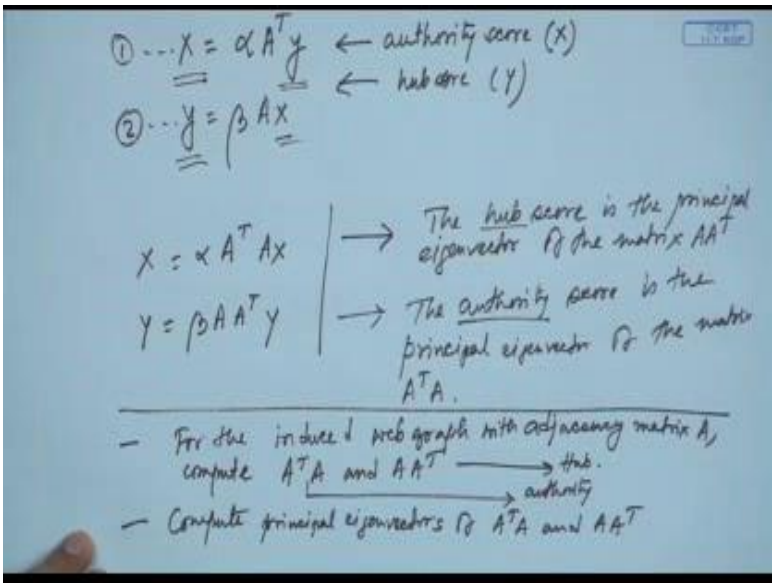
So, it is very simple. Given the directed adjacency matrix you compute $A A^T$ and $A^T A$. Now you compute the principal eigenvector of each of these product matrices and each of them actually represents the hub and the authority scores correspondingly. So now, given these two values the steps of the algorithm are very simple as I have already stated in the slides. So, look at this slide.

(Refer Slide Time: 25:34)



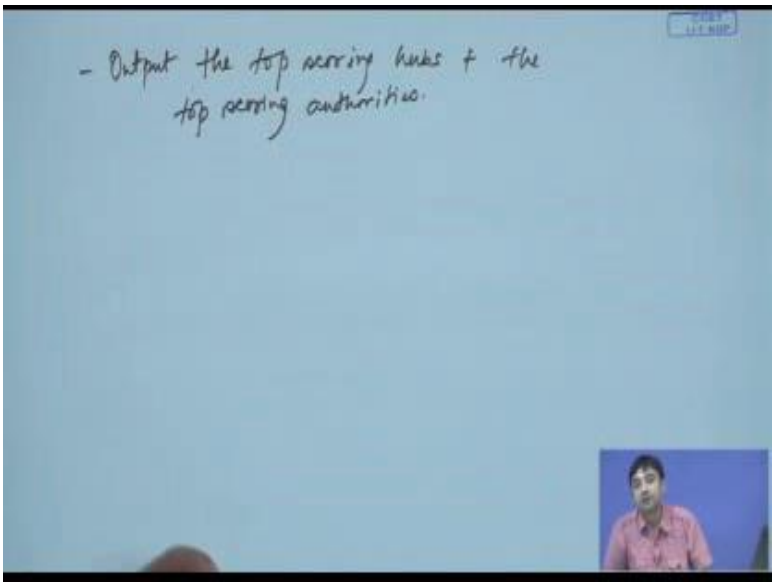
There are basically three steps in the algorithm. Assemble the target subset of web pages from the graph induced by their hyperlinks and compute A A^T and A^T A; there is a printing mistake here one is A^T A and the other is A A^T.

(Refer Slide Time: 25:58)



Basically the first step of the algorithm is, for the induced web graph with adjacency matrix A compute A^T A and A A^T . This actually stands for the authority and this one stands for the hub. This is the first step. Next, compute principal eigenvectors of A^T A and A A^T . So, we compute the principal eigenvectors of these two matrices.

(Refer Slide Time: 26:58)



Now, the third step is output the top scoring hubs plus the top scoring authorities. See this is the interesting difference from the Page Rank algorithm, when we were ranking the nodes based on Page Rank you would just return those nodes which are basically good authorities, you will miss the good hub. So now, we are using this advanced calculations we have now incorporated both the good hubs as well as the good authorities in our search results.

We will stop here.

Thank you.