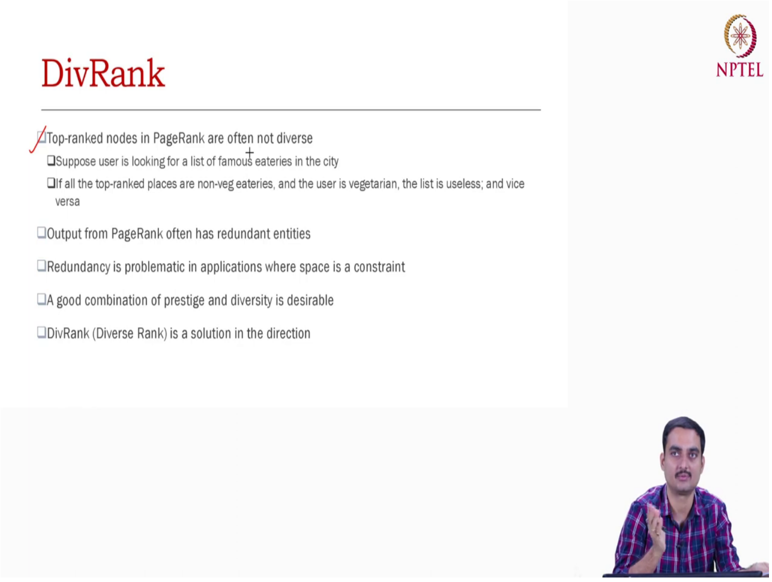
Social Network Analysis
Prof. Tanmoy Chakraborty
Department of Computer Science and Engineering
Indraprastha Institute of Information Technology, Delhi

Chapter - 04
Lecture - 24
Lecture - 05

So, since we you know understood what is PageRank and how PageRank works. And since we have already you know understood one of the problems in PageRank that it is not able to capture the diversity of a network. Now, we discuss something called DivRank ok.

(Refer Slide Time: 00:41)



DivRank

- ❑Top-ranked nodes in PageRank are often not diverse
  - ❑Suppose user is looking for a list of famous eateries in the city
  - ❑If all the top-ranked places are non-veg eateries, and the user is vegetarian, the list is useless; and vice versa
- ❑Output from PageRank often has redundant entities
- ❑Redundancy is problematic in applications where space is a constraint
- ❑A good combination of prestige and diversity is desirable
- ❑DivRank (Diverse Rank) is a solution in the direction

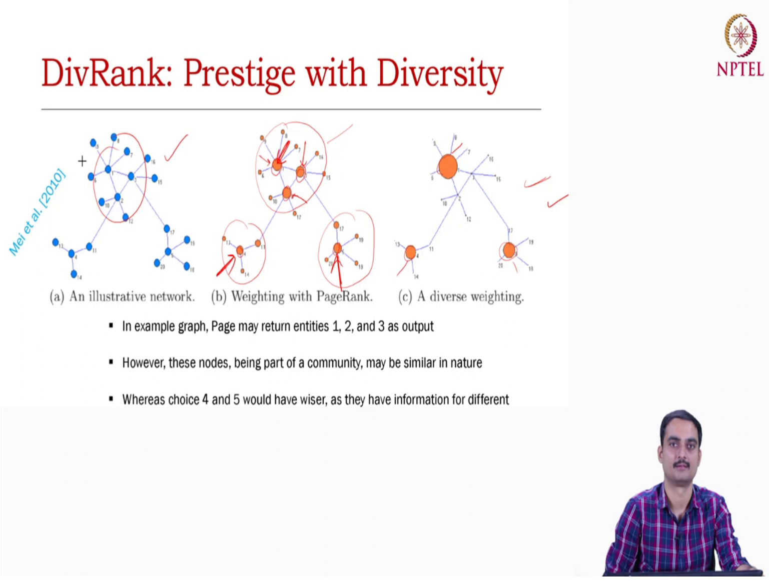
So, diversified PageRank or whatever diversified rank. So, in case of PageRank we already mentioned that top ranked nodes in a PageRank are often not diverse right. And diversity is important. For example, say you know you are in a city and you want to explore different restaurants for example, right. And you do not have any such preference, but you prefer that I mean whenever you search some search about a restaurant in the city the top ranked restaurant should come right. And what would be your expectation your expectation would be a best ranking system should be able to produce a rank where the top few restaurants would have diverse recipe.

For example, one restaurant is famous for it is your chicken related items right, other restaurant is famous for its more of a continental type of items right, other restaurant is famous for its you know veg items vegetable items right mostly veg restaurant. So, you expect that in the top 5 6 ranks you should not have in redundancy. Meaning that you should not be shown restaurants which are famous for only for chinese foods for example, or only for continental foods right.

They should have they should have a diversity in the top ranks right. But what happens in case of PageRank is that PageRank always returns restaurants which are popular right. Irrespective of whether they are diverse or not ok. So, therefore, diversity is important. And you know and diversity is important because we generally want to reduce the redundancy in the ranking process right.

So, a good way to tackle this problem is that you make a trade-off between prestige and diversity. So, prestige can be captured using PageRank. So, can we come up with some sort of tradeoff which gives equal weightage or equal importance to both prestige and diversity. So, DivRank diverse rank or diversified rank is one of such solutions ok.

(Refer Slide Time: 03:15)



DivRank: Prestige with Diversity

(a) An illustrative network. (b) Weighting with PageRank. (c) A diverse weighting.

- In example graph, Page may return entities 1, 2, and 3 as output
- However, these nodes, being part of a community, may be similar in nature
- Whereas choice 4 and 5 would have wiser, as they have information for different

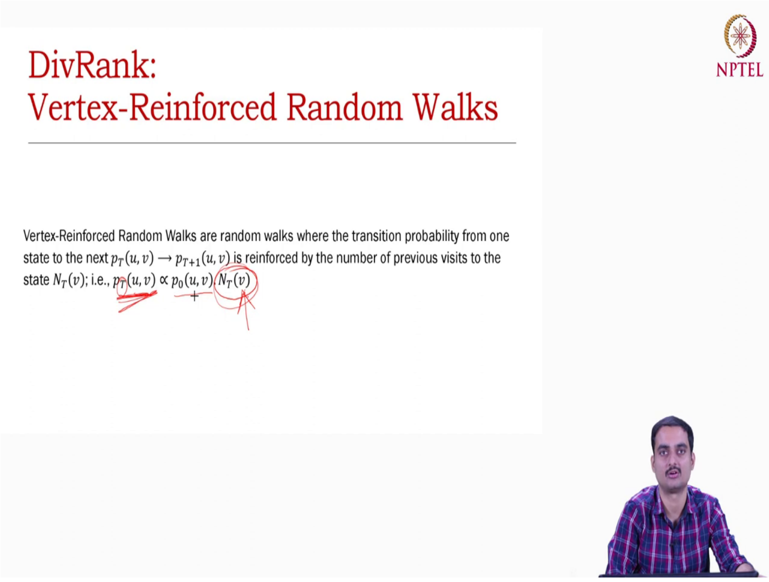
So, here is a graphical example that I briefly gave in the last lecture. So, let us say this is the network ok. And in this network on this network if you run PageRank what would happen? That node 2, node 3 and node 1 these three nodes will be returned as top 3 nodes ok. But you may not want to you may not want to return all these three nodes because all these three nodes basically represent this component of the network. Rather, what you want? You basically want that some node from this component would also be returned some node from this component will also be returned and one node from this component can be returned.

So, this is exactly the DivRank does. DivRank could return node 1 node 4 and node 5. So, this is DivRank's output node 1 node 4 and node 5 as top 3 nodes here the size of the node corresponds to the prestige. So, in case of PageRank you see that these nodes are of equal size right followed by these nodes whereas, here you see that these nodes are top ranked nodes. Others nodes do not have any significant rank at right.

So, what DivRank does? DivRank kind of allows nodes to compete with each other. There is no collaboration there is no cooperation. In case of PageRank there is a cooperation because ultimately prestige flow flows from 1 node to another node and you basically let the you basically allow the node to collaborate with each other to increase each other importance. Here you basically allow the nodes to compete with each other ok.

So, if you look at this portion of the network what happens is that due to the competition a node which you know defeats other nodes that particular node would keep on you know absorbing importance from the remaining neighbors of the node and the importance or the prestige would basically go on and on go increase I mean go on increasing over time; that is the idea ok.

(Refer Slide Time: 05:37)



DivRank:
Vertex-Reinforced Random Walks

Vertex-Reinforced Random Walks are random walks where the transition probability from one state to the next $p_T(u,v) \rightarrow p_{T+1}(u,v)$ is reinforced by the number of previous visits to the state $N_T(v)$; i.e., $p_T(u,v) \propto p_0(u,v) N_T(v)$

So, now how do we implement this how do we materialize this idea? So, DivRank is based on something called vertex reinforced random walk. What is vertex reinforced random walk? Basically the idea is that you know whether you visit a particular restaurant or whether you visit a particular museum right depends on the number of times that restaurant or that museum you know have been visited in the past.

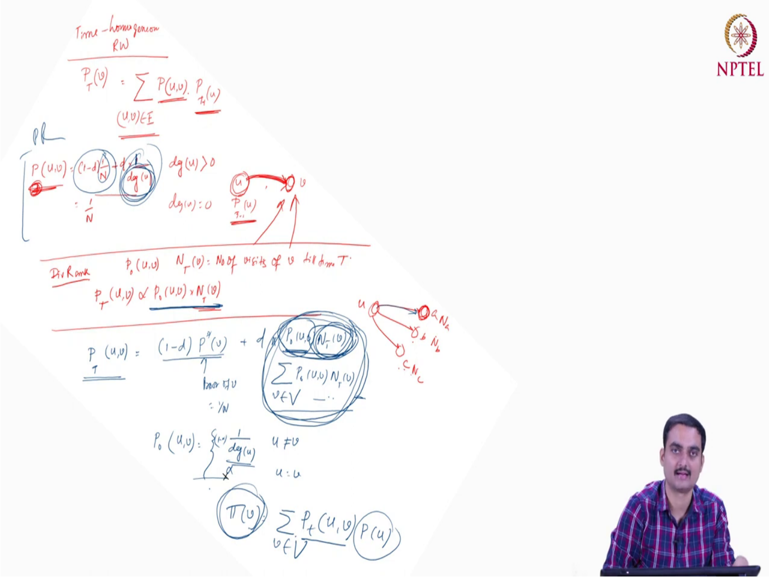
So, the more a node has been visited again and again. The more I mean the prestige of the node would increase more and more over time. So, the transition from node u to node v depends on node v’s previous visits how many times node v has been visited by the random walker ok. That is the that is the cracks of the of the theory here of the of DivRank. So, here there is a notion of time heterogeneous random walk right.

Meaning that this jumping probability from u to v. So, far if you remove the PageRank algorithm right the transition probability from u to v that remains same and that is always uniform right. I mean uniform in the sense like given a particular node if there are four forward nodes the probability of choosing one of these forward nodes is uniform. It does not depend on the distance the other node the other part of the node in the edge right.

Whereas in vertex reinforced random walk the idea is that the transition probability that keeps on changing over time it is not static and it is dependent on the number of times that you have visited the destination node or the end node of the edge. So, if there is an edge from u to v right and this probability depends on time right. So, as the time progress what would happen is that this probability would change with respect to N T v. What is N T v? N T v the number of visits of node v at time T.

And this is dynamic. See this is dynamic this is also dynamic. You have some initial probability initial transition probability which is same as the PageRank kind of probability, but over time this would increase or whatever decrease depending on this right. So, this is called vertex reinforced random walk. You are reinforcing based on the number of visits you know of the particular node ok. So, let us let me first use my white board and discuss some of the important components.

(Refer Slide Time: 09:14)


Time-homogeneous RW
$P_T(v) = \sum_{(u,v)\in E} P(u,v) \cdot P_{T-1}(u)$
PR
$P(u,v) = (1-d)\frac{1}{N} + d \times \frac{1}{dg(u)}$ $dg(u) > 0$
$= \frac{1}{N}$ $dg(u) = 0$
u
v
$P_{T-1}(u)$
DivRank $P_0(u,v)$ $N_T(v)$ = No of visits of v till time T
$P_T(u,v) \propto P_0(u,v) \times N_T(v)$
$P_T(u,v) = (1-d)\, P^*(v) + d \frac{P_0(u,v) N_T(v)}{\sum_{v\in V} P_0(u,v) N_T(v)}$
prior of v $= 1/N$
$P_0(u,v) = \begin{cases} (1-\alpha)\frac{1}{dg(u)} & u \neq v \\ \alpha & u = v \end{cases}$
$\pi(v) = \sum_{v\in V} P_T(u,v)\, P(u)$
u a N_a b N_b c N_c
NPTEL

So, let us look at you know in case of traditional PageRank right. Time homogeneous random walk. You know why this time homogeneous? Because the transition probability does not depend on the time. So, in case of time homogeneous random walk the probability of a node v or prestige of a node v right. This is not probably this is not probability this is basically prestige or whatever PageRank whatever right.

This is what this is essentially sum of P of u v times P of t u. If there is an edge from u to v ok. So, this is v this is u ok. So, you have so, sorry this is T minus 1. So, at T minus 1 is time T minus 1 is theta iteration node u has some prestige and that is P T minus 1 u right.

Now, that would be that would basically move through this edge which is this probability P of u v now P of u v in case of PageRank P of u would be1 by the number of forward edges of node u right something right is a is some probability. So, the PageRank or whatever prestige of v is this probability times this 1 similarly you have other types of nodes ok. And you sum them up. This is time homogeneous.

So, how do you define this P of u v as I mentioned P of u v is basically can be in two ways if the degree of u is greater than 0 and if the degree of u is equals to 0. If the degree of u is equal to 0 means it is a dead end u does not have any outward edge forward edge then it would be 1 by n. If this is not a dead end then normal PageRank equation d into 1 by degree of u right.

So, this would be multiplied with this one to get the page rank, but this is just a transition probability P u v. There is no T factor here remember this. So, this is homogeneous ok. So, in case of DivRank in case of DivRank what we say is that you know let us start with some initial transition probability P 0 u v and this is some kind of prior belief or prior reinforcement whatever right.

And what is N T v? N T v is the number of visits of v node v till time t ok. And the idea is that the probability this transition probability u comma v at time t, this should be proportional to P 0 u v times N T v. Meaning, say this is u this is a b c right in the PageRank you can choose 1 of these nodes a b c uniform random and you jump, but in case of DivRank, it depends on the visiting probability N a, N b and N c say for example, this node is visited has been visited multiple times.

So, it is highly likely that the random walker basically move from u to a not to b not to c ok and this is the idea. Think about it what is happening here? So, for a random walker basically the random walker you know keeps on moving to such nodes which have been visited by the random walker in the past multiple times ok. So, in turn it basically absorbs on the you know prestige from its neighborhood nodes its neighboring nodes ok because of this reinforcement ok. So, now, let us look at the general form of DivRank.

So, P T u v right. So, initially we have this one for in case of PageRank. Now, we define a time heterogeneous random walk P T u v this would be some sort of 1 minus d plus d ok. You have some prior about v P star v this is prior probability prior of v. This can be 1 by N if all the nodes are equally likely right. But you already know let us say you already know that this set of nodes are more likely to be visited independent of their independent of their previous visits you know that ok these nodes the these restaurants are owned by say TATA or you know all these you know this this business guy Oberoi and so on.

So, they would automatically attract audience ok. So, therefore, for those nodes you can give more prior for other nodes for other nodes you can give less prior ok. So, this is same as this part now here is the tricks. So, d times now this is not uniform ok. Now you replace this by this. So, earlier this was 1, 1 by degree now this is P 0 u v times N T v summation of P 0 u v N T v over all the v in the graph ok.

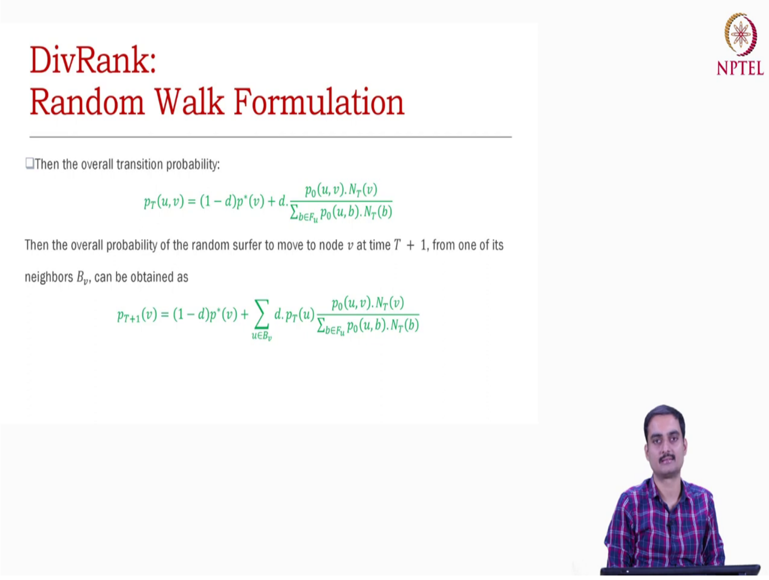
So, what it is saying? It basically says that the jump probability the random jump the jumping probability this transition probability is proportional to this one. In other words this is proportional to this one because this is fixed this is fixed. You can fix it ok. And the denominator is basically useful because you wanted to make it as a probability. So, denominator is basically same as is basically sum of this quantity across all the nodes.

So, that this part would be a probability ok. This is your new transition probability. Now you can define P 0 this P 0 u v right I said this is constant right this is kind of a constant because this is not independent of time it can be I mean you can use may I mean you can basically incorporate the degree information here as well. Because so far you have seen that degree information degree of node u has not been considered anywhere right. Here degree of u was considered, but here degree of u has not been considered.

So, you can consider P 0 and you can add the you can incorporate the degree information degree of u in some ways you can say that ok let us say if u not equals to v ok. It is basically 1 by degree of u right. If u equals to v then you have some alpha say this is 1 minus alpha ok. This can be a prior. Now if this is a prior then the prior is same as the prior that you use in case of PageRank ok. So, the main difference is this part this component ok. So, ultimately your final DivRank right is essentially what is essentially this one.

So, all the probabilities ok. So, you at time T you look at the transition probability from u to v. So, you want to you are computing the final DivRank value of node v. So, you look at all the edges moving towards v right. So, P 0 sorry P T u v, P T u v all the u v edges right at time t all the inward edges right at time t and you multiply it with the prestige of u. That would give you the final ranking ok. Now, this is the idea behind DivRank.

(Refer Slide Time: 19:56)



DivRank:
Random Walk Formulation

❑Then the overall transition probability:

$$p_T(u,v) = (1-d)p^*(v) + d.\frac{p_0(u,v).N_T(v)}{\sum_{b \in F_u} p_0(u,b).N_T(b)}$$

Then the overall probability of the random surfer to move to node v at time $T + 1$, from one of its neighbors B_v, can be obtained as

$$p_{T+1}(v) = (1-d)p^*(v) + \sum_{u \in B_v} d.p_T(u)\frac{p_0(u,v).N_T(v)}{\sum_{b \in F_u} p_0(u,b).N_T(b)}$$

So, in other words in short if I explain this you know it basically gives you a way to way to a node to compete with each other to compete with its neighbors right. And if you somehow are able to attract the random walk process or the random walk to towards you again and again or say if you if the museum somehow you know has been able to attract audience in the last few months or last few years it is highly likely that in the next year it would again attract more audience.

Same for restaurant right and this and remember this is this does not depend on who is pointing you of course, it is dependent on who is pointing you, but it also depends on your own capability. Say you are not that old restaurant right, but you have created your restaurants with lot lots of lot lots of you know passion and you are pretty sure that you know it would immediately attract people right. It does not need to wait for you know 1 month or 2 months or 1 year right.

That kind of flexibility is not provided in case of PageRank, but that is provided in case of DivRank ok. Now DivRank is not only used for web page ranking it can also be used for graph summarization graph summary generation. For example, in case of graph summary what we are interested in, we are interested in returning nodes or sub graphs of a node a sub graph of a graph which are diverse, but prestigious as well right.

So, you want to choose those sub graphs or those nodes which are far apart from each other, but prestigious ok. Prestige also important. So, the these two things are very you know nicely balanced in case of DivRank. So, I stop here. In the next lecture we discussed other we will discuss other kind of metrics like SIM rank Metapath meta path and path SIM and so on and so forth ok. So, we stop here.

Thanks.