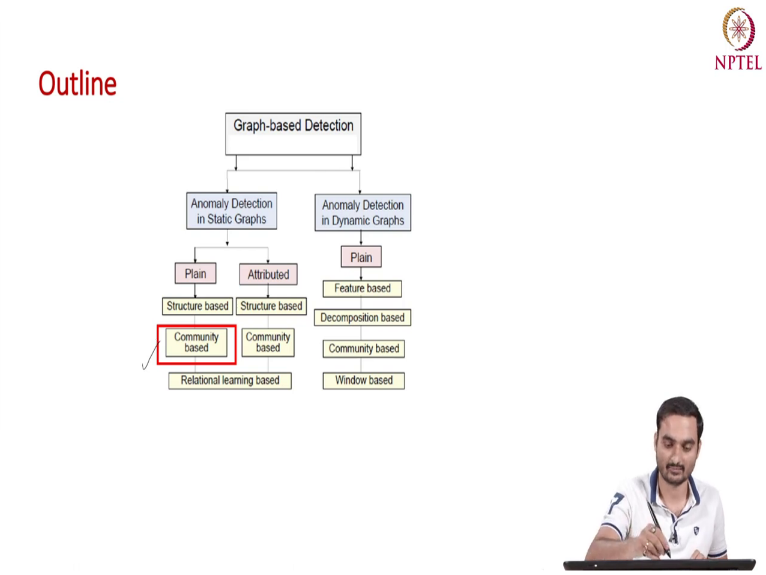
Social Network Analysis
Prof. Tanmoy Chakraborty
Department of Computer Science and Engineering
Indraprastha Institute of Information Technology, Delhi

Chapter - 08
Lecture - 03

So, in the last lecture we have started discussing on algorithms right, for detecting anomalous entities from a graph, ok. And we looked at you know a taxonomy.

(Refer Slide Time: 00:34)



Outline

Graph-based Detection

Anomaly Detection in Static Graphs

Plain

Structure based

Community based

Relational learning based

Attributed

Structure based

Community based

Anomaly Detection in Dynamic Graphs

Plain

Feature based

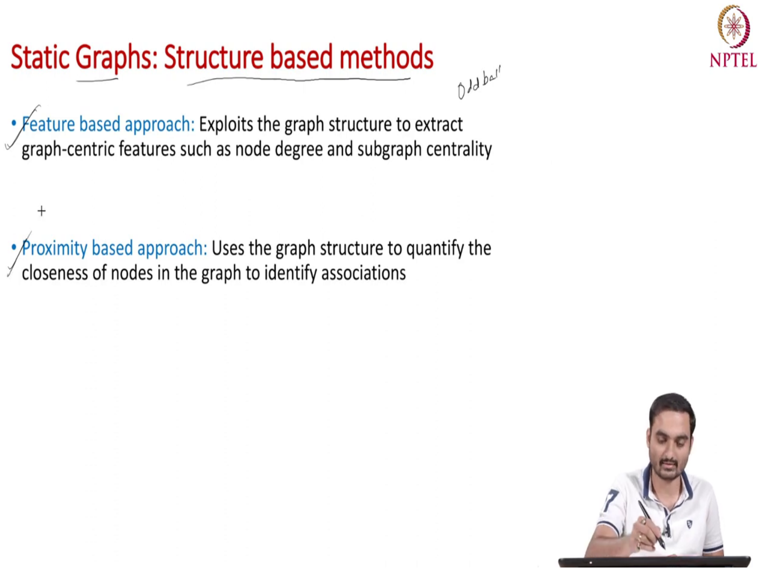
Decomposition based

Community based

Window based

And we have discussed the and we are discussing algorithm where we use a plane graph, right.

(Refer Slide Time: 00:46)



Static Graphs: Structure based methods

Oddball

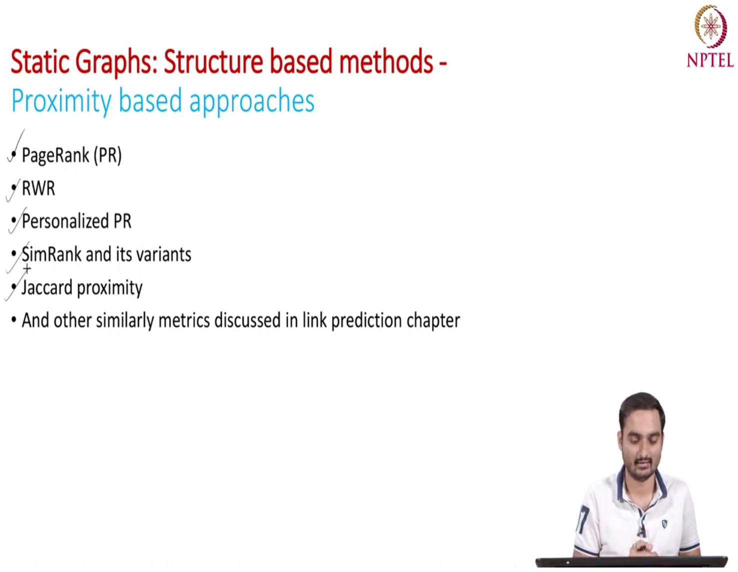
- Feature based approach: Exploits the graph structure to extract graph-centric features such as node degree and subgraph centrality

+

- Proximity based approach: Uses the graph structure to quantify the closeness of nodes in the graph to identify associations

And we use different structural properties right, mostly feature based approach right, to detect anomalous entities. And we have discussed one such algorithm which is called oddball, right. So, in today’s lecture we look at the other types of algorithms. So, we look at proximity based approaches.

(Refer Slide Time: 01:14)



Static Graphs: Structure based methods - Proximity based approaches

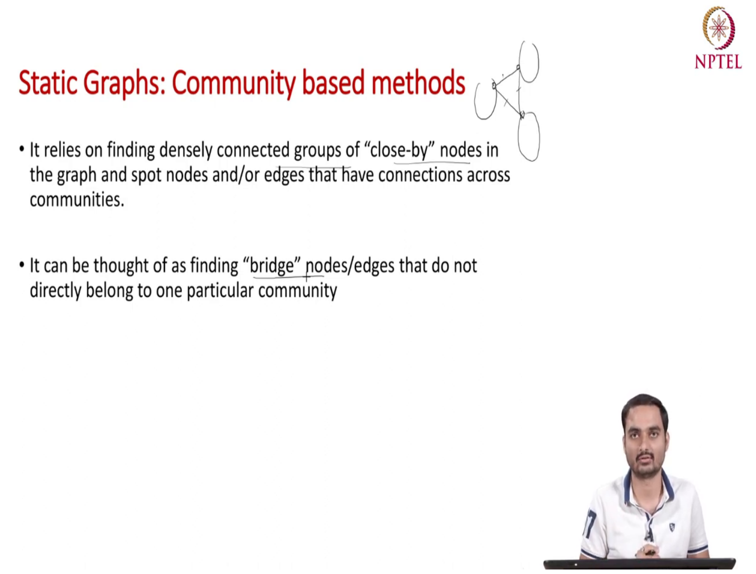
- PageRank (PR)
- RWR
- Personalized PR
- SimRank and its variants
- Jaccard proximity
- And other similarly metrics discussed in link prediction chapter

So, mostly in the proximity based approach we basically measure the distance between a pair of nodes and based on that we essentially compute different matrix. For example, PageRank we have already discussed, random work with restart we discussed in the link analysis chapter, Personalized PageRank we also discussed in the link analysis chapter.

SimRank and it variants also discussed in the link analysis chapter, Jaccard coefficient Jaccard proximity and other simulating matrix for example, say atomic adder right, and say half depression index and authority depression index and so on and so forth. So, all these measures are generally used to check the distance between a pair of nodes, right and if a node is very far from the remaining nodes in the graph.

Then you can say that that node is basically an outlier, ok. So, let us look at the community based approaches as I mentioned I will not go deeper into all such in a possible cases, but I will focus on very very few specific algorithms which have been used as a baselines for quite some time

(Refer Slide Time: 02:30)



Static Graphs: Community based methods

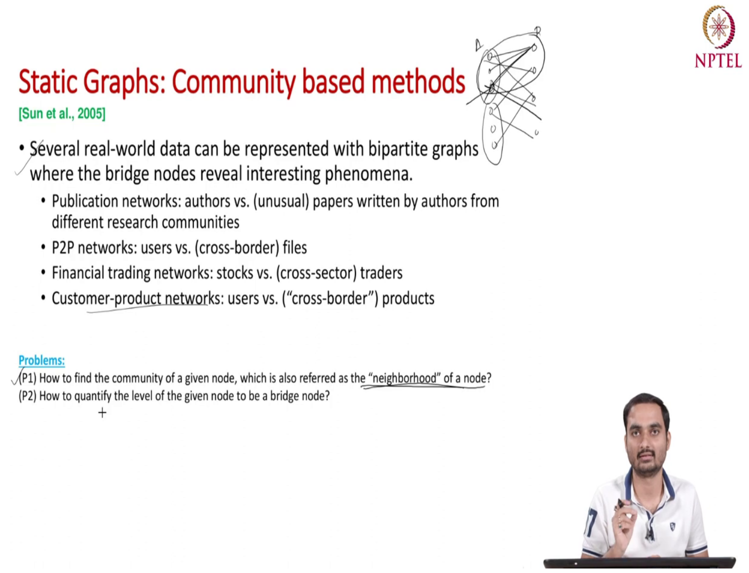
- It relies on finding densely connected groups of "close-by" nodes in the graph and spot nodes and/or edges that have connections across communities.
- It can be thought of as finding "bridge" nodes/edges that do not directly belong to one particular community

So, in the community based methods basically again in the static graph static plane graph ok, non attributed graph. So, basically here the idea is that it like. So, in the static plane graph community based methods the aim is to find densely connected groups of close-by nodes right, in the graph and spot nodes or edges that have connections across communities.

So, it is basically saying that you have you know these communities and then say let us say nodes are like this right, articulation points we discussed earlier. And we have edges which are connecting different communities. And so this edges which are basically b bridge edges those edges can be annotated as can be identified as anomalous edges, right.

So, it can be thought of as finding bridge nodes and edges that do not directly belong to one particular community. Now this is one way of defining an outlier for a given graph and community, ok. You can also define in your own way, right.

(Refer Slide Time: 03:57)



Static Graphs: Community based methods

[Sun et al., 2005]

- Several real-world data can be represented with bipartite graphs where the bridge nodes reveal interesting phenomena.
  - Publication networks: authors vs. (unusual) papers written by authors from different research communities
  - P2P networks: users vs. (cross-border) files
  - Financial trading networks: stocks vs. (cross-sector) traders
  - Customer-product networks: users vs. ("cross-border") products

Problems:

(P1) How to find the community of a given node, which is also referred as the "neighborhood" of a node?
(P2) How to quantify the level of the given node to be a bridge node?

So, now, let us look at some of the algorithms again very briefly. Say you know several real-world data sets can be represented in terms of bipartite network, say and in the bipartite network if you think of you know bridge nodes which again connect different communities different clusters that can be useful. For example, if you think of publication network say author, paper bipartite network and if an author has published a paper you can connect and so on, right.

So, in this particular graph you can possibly see that ok, this is one community where nodes have collaborated quite these authors have collaborated quite frequently. Similarly, there is another community where authors have collaborated frequently and there are authors which is basically connected to this community as well as this community, ok.

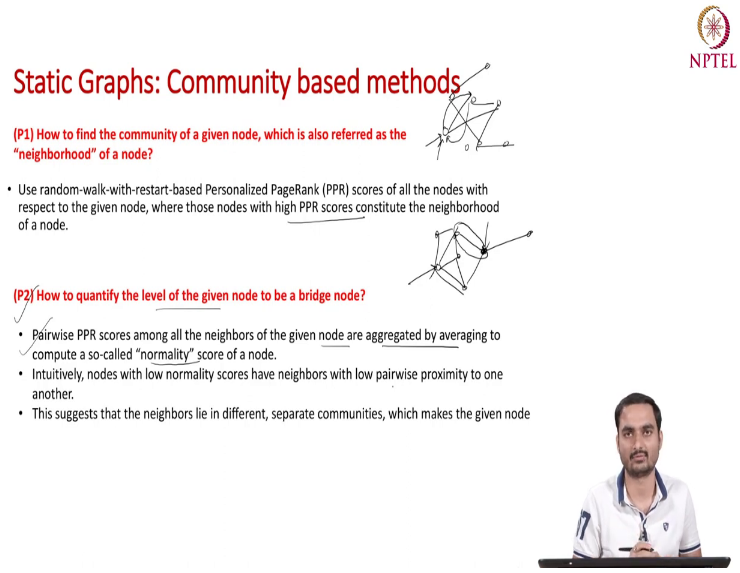
You can think of this author as an interdisciplinary author, right working in diverse areas. Similarly, in the customer product network if you think of user product bipartite network users generally tend to buy products you know based on the interest of their neighbors. So, we when we buy products, we basically ask for recommendation from my from our friends neighbors and sometimes they recommend and based on that we buy products, but if you see cases where users are connected to multiple communities, right.

User is connected to multiple products for example, and the products and say let us say the products are diverse in nature then that particular user may be right, maybe an anomalous user, ok. So, here what are the problems there are two problems. The first problem is, how to find the community of a given node which is also referred to as the neighborhood of a node?

So, in this type of algorithms we generally do not use traditional community based algorithms to detect communities, right. So, what we do? We generally take a graph and take a node and identify its neighbors, right. And we keep on exploring tightly connected neighbors and we group them together.

Now this is a very vague way of detecting communities, but here the idea is that we detect community and outlier at the same time simultaneously, right. So, the first question is, how to find you know the community of a given node? Which we also call as a neighborhood. And then how to quantify the level of a given node to be a bridge node? Ok. So, remember our task is here to detect node which is which acts as a bridge node, ok.

(Refer Slide Time: 07:03)



Static Graphs: Community based methods

(P1) How to find the community of a given node, which is also referred as the "neighborhood" of a node?

- Use random-walk-with-restart-based Personalized PageRank (PPR) scores of all the nodes with respect to the given node, where those nodes with high PPR scores constitute the neighborhood of a node.

(P2) How to quantify the level of the given node to be a bridge node?

- Pairwise PPR scores among all the neighbors of the given node are aggregated by averaging to compute a so-called "normality" score of a node.
- Intuitively, nodes with low normality scores have neighbors with low pairwise proximity to one another.
- This suggests that the neighbors lie in different, separate communities, which makes the given node

Of course, we have discussed many algorithms edge between s node between s and so on and so forth. In the past, but here we look at another such algorithm to identify bridge nodes, ok.

So, the first question how to identify neighborhood of a node? What we do here? We here do something called random-walk-with-restart we discussed it earlier, right. So, given a graph we start from a node, right.

And do random-work-with-restart with certain probability the random walker jumps right, and with certain probability it again comes back to the seed node, ok. And what it would do? This random-walk-with-restart would essentially score all the nodes based on the probability of visiting that nodes, ok. So, in the stationary distribution you will get some sort of PageRank score, right. And this is Personalized because there is a jumping probability and this jump allows you to only move to the seed node, right.

So, you will get Personalized PageRank scores for all the nodes, right. Now think about it. So, then what you do? Then if nodes have higher personalized PageRank values right, those nodes are basically neighbors. So, you repeat this random-walk-with-restart for example, multiple times, ok.

And those nodes which are within the close proximity of the given seed node those nodes will be visited multiple times and those nodes will also gain high Personalized PageRank score, right. So, this is the idea. So, in this way we identify nodes with high Personalized PageRank scores, right. And we basically assign them as neighbors of the given seed node, ok.

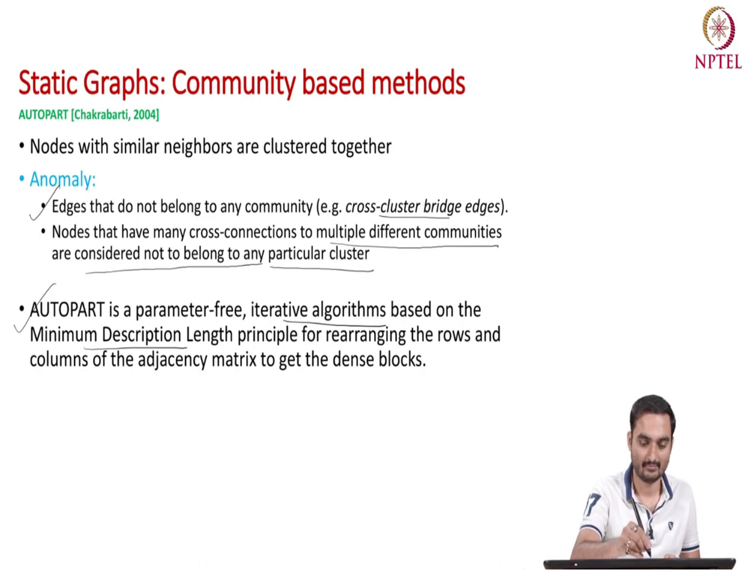
So, the next question is, how do we quantify the level of a given node to be a bridge node? Ok. So, pairwise Personalized PageRank score among all the neighbors right, of the given node are aggregated by averaging to compute so called normality score, ok. So, what are we doing here?

So, we take a pairwise we take a pair of nodes, right. We take a pair of nodes and we measure the pairwise Personalized PageRank scores, right. And if the scores are quite similar then we say that ok, this is a basically a normal kind of nodes, ok. So, intuitively nodes with lower normality scores have neighbors with low pairwise proximity to one other, think about it.

Let us say, let us say ok, you start from the seed nodes, ok. And you repeat the random-work-with-restart multiple times. So, this node will receive less Personalized PageRank value, right. Now if we measure the pairwise similarity of Personalized PageRank scores, right.

Automatically this node will have less pairwise similarity with all the most of the nodes, right. So, therefore, this node can be identified as an outlier node or a bridge node, ok. This is the idea a very simple idea, but of course, you can make it sophisticated, but this is the idea, ok.

(Refer Slide Time: 11:09)



Static Graphs: Community based methods

AUTOPART [Chakrabarti, 2004]

- Nodes with similar neighbors are clustered together
- Anomaly:
  - Edges that do not belong to any community (e.g. *cross-cluster bridge edges*).
  - Nodes that have many cross-connections to multiple different communities are considered not to belong to any particular cluster
- AUTOPART is a parameter-free, iterative algorithms based on the Minimum Description Length principle for rearranging the rows and columns of the adjacency matrix to get the dense blocks.

So, there is another algorithm which essentially you know identified nodes with similar neighbors. So, you know. So, this is the idea. So, edges that do not belong to any community, again the cross-cluster bridge edges right, those edges can be tagged as anomalous edges.

So, nodes that have many cross connections to multiple different communities right, are considered not to belong to any particular cluster, ok. So, this algorithm AUTOPART right, proposed long time back is a parameter free this is a non parametric kind of iterative algorithm which uses minimum description length.

I am not going into details of this, but it basically uses bits right, to encode nodes and the idea is that if a node is rare that node will be encoded with a lengthy bit I mean set of bits, right. So, and this way it basically identifies nodes which are which are rare which are outlier anomalous in nature, ok alright.

So, in the next lecture we will look at the attributed graph and we will discuss one such algorithm in details which takes into account the community information as well as the attributes of nodes and edges to identify outliers or anomalous entities in a network.

Thank you.