

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

Chapter - 05

Lecture - 02

So, in the last lecture we started discussing about community structure and we have seen we basically you know I try to motivate you why it is an important problem and why it is a non trivial problem, ok. So here, we will discuss different types of community structures, ok. There are generally three types of community structures, but sometimes there are you know one more type so, there are essentially 4 types.

(Refer Slide Time: 00:51)

Types of Communities: Disjoint Communities



- Also referred to as **flat communities**
- Each node in the network can belong to at most one community
- Differs from **disconnected components**:
 - nodes in two different communities can still have connecting edges
 - referred to as **bridges**
- Example: Full-time employees of an organization

Disjoint Community Structure
<http://optnetsci.cise.ufl.edu/research/disjoint-overlapping-communities/>



So, the first type is disjoint communities. It is also called as flat communities, ok. What is a disjoint community structure? In a disjoint community structure a node can only belong to a single community. A node can belong to at most one community, at most and at least one community. So, exactly one community a node can belong to, ok. Let us look at this example. Here you see that each community this circles right different circles the circles indicate different communities so; there are 4 communities and you see that a node can belong to only one community, ok.

(Refer Slide Time: 01:33)



Types of Communities: Overlapping Communities



<https://stackoverflow.com/questions/51102350/python-remove-overlapping-communities-in-graph-plot>

- Members can belong to more than one community at a time
- Communities can even share edges
- Realistic and generic community structure
- Harder to find than flat communities
- Example: Various groups in social networks

$f(c)$
2





So, the next one is overlapping community. What is overlapping community? An overlapping community a node can belong to more than one community ok and this is more natural. If you think of social networks right this is more natural you can be a part of say Sachin Tendulkar fan club, you can be a part of Narendra Modi fan club, you can be a part of you know Bollywood movies and so on and so forth, right.

A node can at the same time a node can belong to multiple communities. If you look at here say this node ok, this node is a part of the black community as well as the green community, ok. So, say this node right this one this node is a part of the red community as well as the blue community and so on and so forth. So, disjoint community detection is of course difficult, but overlapping community detection is even difficult, more difficult, why? The reason is that, if you think of an exhaustive way to detect communities, right.

Say you have certain function f ok, a certain function f and you have a network where you have different nodes ok and you know let us assume that you know the number of communities ok. Let us assume that you know that there are there cannot be more than 2 communities, ok. So, what are the possible ways to group nodes into two communities? There are different ways for example; you one grouping would be this ok, another grouping would be this and so on there are many ways right of grouping nodes into 2 communities. What about overlapping community?

So, this is disjoint community. When it comes to overlapping community, the number of possible options will increase exponentially, ok. So therefore, overlapping community detection is even more difficult than a disjoint community detection.

(Refer Slide Time: 03:31)

The slide is titled "Types of Communities: Hierarchical Communities". It features a network diagram on the left with nodes and edges, and a list of bullet points on the right. Handwritten red annotations include "Sp", "E, net", "ECE", and "CS" with arrows pointing to specific parts of the diagram and text. A small inset diagram at the bottom shows three overlapping circles labeled 1, 2, and 3. The NPTEL logo is in the top right corner.

Types of Communities: Hierarchical Communities

- Outcome of merging two or more flat or overlapping communities in a network
- Can be linked to other hierarchical, overlapping, or flat communities
- Example: various city-level communities merged to form a state-level community

<https://www.sciencedirect.com/science/article/abs/pii/S0020025514011463>

The third one is called hierarchical community. In an hierarchical community as the name suggests so, nodes are grouped into different hierarchies. For example, in a student interaction network right say for example, the low level groups right you can think of low level groups as say these are students ok; the first label grouping can be done based on you know based on year.

So, 1st year, 2nd year, 3rd year, 4th year so, this is 1st year student, 2nd year student, 3rd year student and so on, ok. Another the next level grouping can be this 1st year, 2nd year, 3rd year Computer Science right; you have another 1st year, 2nd year, 3rd year ECE, ok. Again another level of grouping can be this CS, ECE, IT right this will form engineering, right. Then, you have you know other science subject Physics, Chemistry that would form science, right.

Similarly, you have Arts; you have Commerce right and then, this Science, Arts, Commerce can again further be grouped into an institute. Within an institute you have Science, Arts Commerce and then you know within again within science you have a different branch within engineering different branch and so on, right. Then, again you can group them based on institutes right and so on and so forth.

So, again hierarchical structure is quite obvious, quite frequent in our day to day world if you think of even biological network right we have see we see cells, protein cells, tissues and so on and so forth, right. So, these are the main community structure types, right.

(Refer Slide Time: 05:26)

Types of Communities: Local Communities



- Shows a community structure from local perspective without focusing on global structure
- Example: citation network formed by research groups inside a university

<https://www.digitaltrends.com/features/the-history-of-social-networking/>



But, sometimes people also say that there is another structure another type of community is called local community. What is the local community? Local community is a community with respect to a particular node. So, let us say the so if somebody ask me tell me the local community of vortex v , right. So, the output would be a community where the node v belongs to, right. So, what do you mean by returning a community, basically I need to return other nodes which are also part of the community.

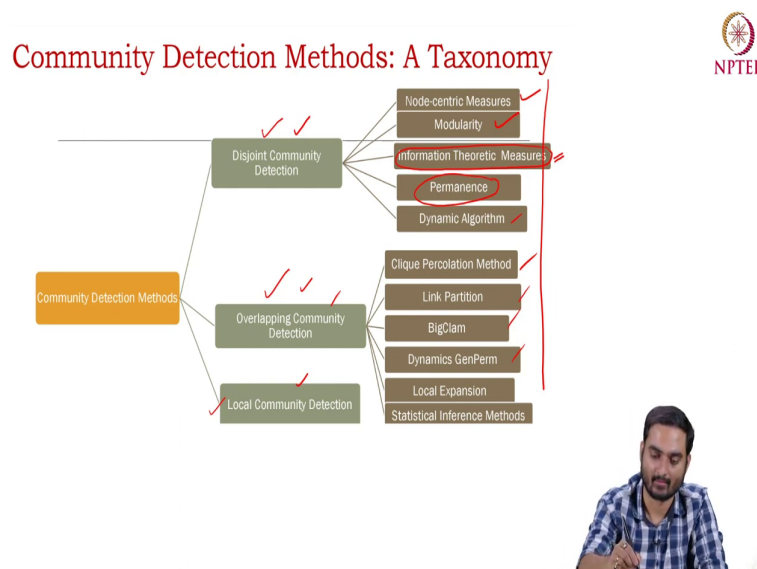
So, I am only interested in that community or that or those nodes which are part of the same community where the node v belongs to ok, right say these are the nodes which are also part of the same community where v belongs to. So, if I if the query is to return the community the local community of node v I will only return right; the nodes which are part of the same community this is called local community. So, local community detection is important you know in with respect to certain applications for example, in a social network it is a gigantic network, right.

And you it is very difficult to detect communities at every day because, network grows over time and you know it becomes exponentially larger and larger over time. So, nobody runs overlapping commutation algorithms or the or non overlapping commutation algorithms from

the scratch what people do is generally when people basically query for a particular vertex, for a particular node and the output should be the community for that particular node.

So, this local community detection is very important in the context of say recommendation system in the context of you know other types of so, you see want to understand, you want to profile a particular user; you want to understand what are the other users who this particular node v interact with most of the times, right. So, we are not interested in other nodes we are only interested in a part of the node, a part of the network which this node interact interacts with quite frequently, ok. So, there the local community based algorithm would be useful.

(Refer Slide Time: 07:48)



So, if you look at the algorithms right. The algorithms for community detection you so this is the; this is the brief taxonomy you see here you broadly you know classify into disjoint community, overlapping community, local community. We generally do not talk about hierarchical community because, hierarchical community structure can automatically be detected during the process of say disjoint community detection we will discuss later that while detecting disjoint communities we also unfold the underlying hierarchy, ok. So, we do not need to explicitly detect hierarchy hierarchical communities, ok.

So, disjoint community overlapping community and local community. So, within disjoint community we will see methods like some node based node centric measures we will see a metric called modularity right. There is another set of algorithms which are based on information theory entropy etcetera. So, this part I will not cover, but if you if you are

interested you can go back and check we will also discuss another metric called permanence which turned out to be highly effective and we will discuss some dynamic algorithm.

For overlapping communities we will discuss clique percolation right link partition big clam and some you know some variations of permanence called jain perm ok and for local communities we will also discuss local expansion some other inference model, ok. So, this is more or less the topics that we are going to cover in this chapter, ok.

(Refer Slide Time: 09:30)

Node-centric Community Detection




- Use the property of the nodes to find community structure in the network
- Exploits node-centric features in a number of ways:
 - Complete Mutuality
 - Cliques
 - Reachability of Members
 - K-cliques
 - K-clan
 - K-club
 - Node Degree
 - K-plex
 - K-core

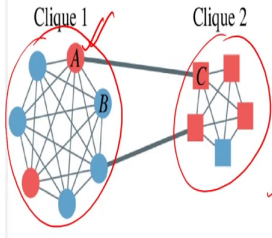


So, let us start with a simple way of detecting communities. Let us look at some of the node centric metrics, ok. So, generally we classify you know these strategies the node centric detection into three groups. One is called complete mutuality right, the second one is something called disability of members and the third one is node degree ok we will discuss all these things one by one. So, let us start with the complete mutuality and this is the; this is the trivial definition of a community, ok.

(Refer Slide Time: 10:03)




Node-centric Community Detection: Finding 'Cliques'



Clique 1 **Clique 2**

- A subgraph of a graph is a clique if every vertex-pair in the subgraph are adjacent
- Has diameter of 1
- Can be considered as communities
- A couple of problems with this approach
 - ✓ Finding cliques from a network is NP-complete
 - Constraints on cliques are too strict a requirement
 - Large cliques are not present in social networks usually

https://www.researchgate.net/figure/illustration-example-of-a-small-sec-clique-network-in-this-example-clique-1-is-a-fig_337025822



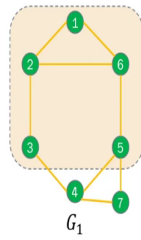
So according to this one, a community is basically a clique, ok. What is a clique? A clique is a completely connected graph, where all the nodes are connected to each other, ok. So, if you think of this one, this part this is a clique because, all the nodes are connected to each other. This is also a clique all the nodes are connected to each other. So, it is a very stricter definition of a community, ok.

So, essentially community detection is nothing but finding cliques in this case ok and it turned out if you see the literature, it turned out that finding cliques from a network is an NP complete problem, ok. So therefore, I mean we generally do not detect cliques we try to you know reduce the restriction of you know connections of all pairs of nodes.

(Refer Slide Time: 11:09)

Node-centric Community Detection: K-Cliques

*maximum
maximal*



The maximal subset of vertices of the network such that, for any two nodes belonging to the subset, the shortest distance between them is less than or equal to K

1-clique is normal clique

The nodes {1, 2, 3, 4} forms a 2-clique in the network G_1

2-cliques are known as group of friends in social network analysis

Issue:

A node not present in K-clique can contribute in formation of the shortest distance in it!

(a, b, c, d, e, f, g) f(a, b) = 0.8
(a, b, c) f(a, b, c) = 0.8
f(a, b) = 0.8
f(a, b, c) = 0.8



So therefore, we relax the restrictions in some ways and then we come up with I mean we can think of different definitions for example, this one. So, this is called K clique. So, we are not interested to detect cliques. We are interested to detect something called K cliques K cliques. What is the K clique? A K clique is the maximal subset of vertices. So, please read this definition carefully, ok.

A K clique is the maximal subset of vertices of the network such that, for any two nodes belonging to the subset, the shortest distance between them is less than equals to K. There are two conditions here. The first condition is, it should be a maximal subset, the second condition is between any pair of nodes in the subset, the distance the shortest path distance should be less than equals to K, K or less than K, ok.

So, let us try to understand this slowly. What is what do we mean by maximal subset? I hope you know the difference between something called maximum and something called maximal, ok. What is the definition? What is the difference? Let us say, let us say you have a set ok, you have a set you have some number you have some numbers or some you have some entities right ok and you have certain function.

This function is defined on a set of set of elements in this set ok and you want to maximize this function, ok. So, you see that when I do not know what is the function, let us assume that is a function and this function takes into account different elements from the set and then it returns some value, ok. And our task is to get you know the maximum value of this function.

So, the task is to identify those elements which you input to this function will return the maximum value, ok.

And let us say, I when I add a, b right I get the maximum value. Let us say the maximum value is 0.8, ok. When I add a, b when I add c here also I also get the same value 0.8, ok. So, this a, b this subset a, b is not maximal it is not maximal why, because, if you add another number another element c it gives it still gives the maximum value, ok. Let us say, a, b, c now gives the maximum value right and if you encounter situation that after a, b, c whatever element you add the value will decrease. Whatever element you add here to the function the value will decrease, then you stop, right.

So, your set that you return is a, b, c. So a, b is not maximal, but a, b, c is maximal this is maximal, ok. So now, let us see let us think of another you know another solution say you added e, f right you got 0.8, ok. You again add d, you still got 0.8. You add c; you still got 0.8, right. But, then you added then you start adding the other elements the value will decrease, ok, so you freeze this. So, your another maximal element is e, f, d, c, ok.

So, this is one maximal, this is another maximal, right. These are maximal because, if you add another element it will not give you the maximum value, it will not give you the best value that you can obtain from the function, these are maximal. Now, what is the maximum value? What is maximum set? The maximum set is this one why, because in this particular set you have 1, 2, 3, 4, 4 elements, but in this set you have only 3 elements. So, in terms of the size this is the maximum.



So, there can be multiple maximals, but there should be always of course one or more than one maximum, but it should be unique, right. So, in this case the maximum set of the set size would be 4. Say let us say, there is another set right there is another subset say a, b, d, e, g right this also produces the same output 0.9. In that case this is maximum why, because the size is 5. But, let us say there is another set write a, b, c say a, b, d, f this also produces 0.8.

So, this and this will be maximum because, there is no set there is no subset of size 5 which satisfies the condition, ok. So, this is the difference between maximum and maximal, ok.

(Refer Slide Time: 17:04)

Node-centric Community Detection: K-Cliques

- The maximal subset of vertices of the network such that, for any two nodes belonging to this subset, the shortest distance between them is less than or equal to K
- 1-clique is normal clique
- The nodes {1,2,3,5,6} forms a 2-clique in the network G_1
- 2-cliques are known as known as friend of a friend in social network analysis
- Issue:
 - A node not present in K-clique can contribute in formation of the shortest distance in it!



Now, let us again look at this definition, it basically says that I want to detect; I want to detect a maximal subset of vertices ok such that, for every pair of nodes in that subset the shortest path distance should be less than equals to K, ok. Let us think of a clique. So, this is a clique right all the nodes are connected. What is the K value of this clique? If it is a K clique, what is the K value of the clique? Shortest spot distance is one between any pair of node, between any pair of node the shortest path distance is 1 always.

So, this is 1 clique, right. Now, let us look at this figure, right. So, you have 1, 2 forget about this square ok this is one boundary just focus on the network. So, you have 1, 2, 3, 4, 5, 6, 7 there are seven nodes right and let us group this 5 nodes. So now, tell me whether this is a if this is a K clique, if this is a K clique then what is the value of K?

So, if you look at the distance right, if you look at the pairs every pair you take any pair, ok. Let us say this one this one, this one this one, this one this one the distance is always 2 or less than 2. If you take 1 and 2 distance 1 maximum distance is 2 shortest path, right. You may ask how is it so, because if you look at 3 and 4, 3 and 5. The distance is 1, 2 and 3 no, the distance is 2 because there is a; there is node 4, right; through 4 it has distance 2, but although 4 is not a part of this group does not matter.


It basically says that, let us take a subset right it has it does not say that I when I calculate the shortest path. I only look at those edges which are present within the group look in the definition it is not mentioned. So, it is not mandatory that when I calculate the shortest path I

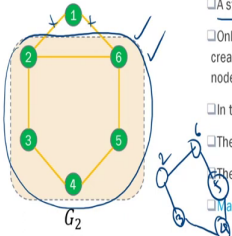
only look at those edges which are part of this group, ok. I can take these edges which are part of which are not part of the group, ok. So therefore, this is a 2 clique ok. What is the problem here?

You may have already identified the problem; the problem here is when we calculate shortest path, right. We may consider nodes and edges which are not part of the shortest not part of this particular group that is wrong, right. So, there should be a restriction on the edges or nodes which we are considering while grouping nodes, right.


(Refer Slide Time: 19:51)

Node-centric Community Detection: K-clan





- A stricter version of K-clique
- Only the nodes present in the set under inspection are used to create the subgraph in which the distance between any two nodes should be less than or equal to K
- In the network G_1 ,
 - The nodes {1,2,3,4,5,6} forms a 2-clique, but it is not a 2-clan
 - The nodes {2,3,4,5,6} forms a 2-clan in the network G_2
 - Maximality condition of K-clique also persists in K-clan



So, in order to address this problem the second definition was is proposed it is called K-clan the names are quite you know ambiguous so, please remember the differences. So, we have clique a relaxed version of clique called K clique right and then we have K-clan, what is this K-clan? So, it is a; it is a stricter version of K-clique, ok. The definition is more or less same.

Only difference is that when we calculate the shortest path we only take into account those edges which are part of this group. So the definition says that, it is basically a maximal subset of nodes right within which if you create the induced sub graph based on the nodes based on only the nodes which are part of the group you create the induced sub graph. So, when I say that induced sub graph, I am discarding those edges which are not part of this group, right. So, for example in this case I am discarding this edge and this edge. So, I only take; I only take this structure 2, 6, 3, 5 and 4, ok.

So, it is a maximal subset of nodes such that, if we take the induced sub graph right of these nodes, right. The distance between any pair of nodes should be less than equals to K , right. Now think of this case, right. If you take any pair of node from this particular group; in this particular group you see that the distance is always 2. So, it is a 2 clan, but this is not a 2 clan, right.

(Refer Slide Time: 21:53)

Node-centric Community Detection: K-Cliques

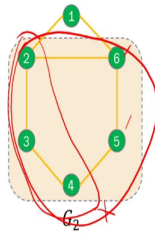
- The maximal subset of vertices of the network such that, for any two nodes belonging to this subset, the shortest distance between them is less than or equal to K
- 1-clique is normal clique
- The nodes {1,2,3,5,6} forms a 2-clique in the network G_1
- 2-cliques are known as friend of a friend in social network analysis
- Issue:
 - A node not present in K-clique can contribute in formation of the shortest distance in it!

And the reason the reason is simple because when I take the induced sub graph right I will not consider this edge this edge and this node, right. If I do not consider this one then, we will have only this part and between 2 between 3 and 5 the distance is 3. So, this would be 3, this would be 3 clan because, the distance between any pair of sub pair of nodes is always less than equals to 3 in this case, ok. So, a stricter definition of this one is K clan. So K clique, then K clan, ok.

So, what about let us take you know another case, let us take another counter example. Let us see let us take let us take this one 1, 2, 3 right 4, 5, 6 right let us take all of them together. Is this a 2 clique, is this a 2 clan? This is not a this is not a 2 clan, right. Because, the distance from 1 to 4 is 3, right.

(Refer Slide Time: 23:27)

Node-centric Community Detection: K-clan



- A stricter version of K-clique
- Only the nodes present in the set under inspection are used to create the subgraph in which the distance between any two nodes should be less than or equal to K
- In the network G_1 ,
- The nodes {1,2,3,4,5,6} forms a 2-clique, but it is not a 2-clan
- The nodes {2,3,4,5,6} forms a 2-clan in the network G_2
- Maximality condition of K-clique also persists in K-clan

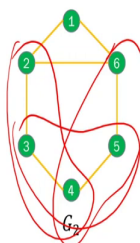


What about this one? What about this grouping say 2, 3 and 2, 3, 4, 5? Is this a 2 clan? If you look at the distance or say let us say this is wrong example let us think of this one say 2, 3, 4. Is this the 2 clan? You take if you take any pair of node the distance is 2 less than goes to 2, you may say this is 2 clan; this is not a 2 clan. Why? Because this is not the maximal set, this is not the maximal set.

If you add 5 and 6 if you add 5 and 6 the set will increase, but it still remains a 2 clan it still remains a 2 clan remember this term. So, maximality should be preserved and the distance constant should be preserved, ok.

(Refer Slide Time: 24:24)

Node-centric Community Detection: K-club



- K-club is a K-clan minus the maximality condition
- {2, 3, 4}, {3, 4, 5}, {4, 5, 6}, {5, 6, 2}, and {6, 2, 3} in G_2 are all 2-clubs
- Every K-clan is a K-club as well as a K-clique
- Challenges:
 - These algorithms are still computationally expensive for large K
 - Deciding appropriate K is difficult



So now, let us look at another definition called K club. So, these definitions are little you know ambiguous, but interesting. What is K club? So, K club is same as K clan, but if we remove the maximality condition, then it is basically K club. So, K clan minus the maximality condition equals to K club. Basically what I mean to say is that a K club is a subset of nodes whose induced sub graphs if you take the induced sub graph of those nodes the distance between any pair of nodes should be less than equals to K.

So, in that case this is a K club right this is also K club this is also K club, but these are not K clan, now think about it. So, K clan is the strict is the most strict is the strictest definition of the most rigid definition of a community, right. If you just ignore clique. So, K clan is the rigid definition then, our relaxed definition is K club where we relax in terms of maximality and another relaxed definition is K clique where we relax in terms of the induced sub graph property, ok.

(Refer Slide Time: 25:55)

Node-centric Community Detection: K-plex

A subset of vertices S in a graph is a K -plex if every vertex of the induced subgraph $G[S]$ has degree at least $|S| - K$.

A measure based on the degree of the nodes

- In the network G_2 ,
- The subset $\{3,4,5,6\}$ is a 1-plex, i.e., a regular clique
- The subset $\{1,3,4,5,6\}$ is a 2-plex, but not a 1-plex
- The subset $\{1,2,3,4,5,6\}$ is a 3-plex, but not a 2-plex

$|S| - K$


So, another there is another notion called K-plex, ok. What is K-plex? K plex is a subset of vertices s in a network such that if every vertex of the induced sub graph. So, induced sub graph constant is there, right. So, I take the subset S , I draw the induced sub graph right and then I look at the degree of every node the degree should be less degree should be at least $|S| - K$ $|S| - K$ is the size of the set that we choose and K is the K plex K , think about it.

You choose a node, you choose a set of nodes S , then you create the induced sub graph $G[S]$, right. Then, you see the you then you basically you know measure the size of the set S , then you look at the degree of individual nodes right and based on that you decide whether it is a K plex or not let us take an example. So, say 3, 4, 5, 6, ok. So, I take the induced sub graph. So, the induced sub graph would be this one 3, 4, 5, 6, ok.

And then, what is the; what is the size of S size is 4 and what is the what is the minimum degree because at least right what is the minimum degree. So, every node has degree 3, right. So, 4 minus so, then it would be 1 plex 4 minus 3 right let us take another example.

(Refer Slide Time: 28:10)

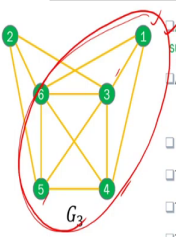
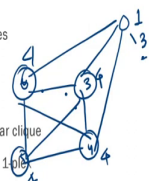
Node-centric Community Detection: K-plex




A subset of vertices S in a graph is a K -plex if every vertex of the induced subgraph $G[S]$ has degree at least $|S| - K$

A measure based on the degree of the nodes

- In the network G_3 , $|S| = 5 - 3$
- The subset $\{3, 4, 5, 6\}$ is a 1-plex, i.e., a regular clique
- The subset $\{1, 3, 4, 5, 6\}$ is a 2-plex, but not a 1-plex
- The subset $\{1, 2, 3, 4, 5, 6\}$ is a 3-plex, but not a 2-plex





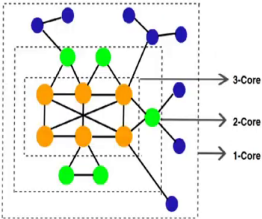


Let us take 1, 3, 4, 5, 6 this one, ok. So, 1, 3, 6, 4, and 5, ok. So, the S is 5 ok and what is the minimum degree? The degree here the degree is 3, degree is 1 to 4, this is 1, 2, 3, 4, this is also 4 this is also 4. So, node has at least degree 3. So, this would be 5 minus 3 plex meaning 2 plex, ok. So, you can relate this with something called core periphery structure that we have discussed earlier, right.


(Refer Slide Time: 28:58)

Node-centric Community Detection: K-core





- Another degree-centric measure
- A subgraph G' of a graph G in which each node has degree greater than or equal to K
- $K+1$ core subgraph can be created from the current K core subgraph by recursively removing nodes of degree K .
- This above should be repeated until there is no node of degree K in the current subgraph.
- Issues:
 - Checking whether a given network is K -core or K -plex is computationally easy
 - Finding maximal K -core/ K -plex is NP-complete!!



So, here we go. So, we have the next definition which is K-core. So, K core we have already discussed earlier, ok. So, the idea behind K core decomposition is that we discussed in the network measure chapter, in chapter 2 I guess. Where we discussed that, we I mean what we do in this case we first remove nodes with degree 0, we keep on removing nodes with degree 0. And then, those nodes which have been removed those nodes form you know the first periphery right or core one. Then you keep on moving nodes with degree 1 until unless you see there is no node with degree 1.

Then, that would form in a core 1. Then you keep on using nodes with degree 2 and so on and so forth. So, core 0 core 1 core 2 core 3 and so on. You see here all these blue nodes right they form core 1, then this green nodes they form core 2 these yellow nodes form core 3 and so on and so forth, ok. So, these are the node centric metrics that we generally use for community detection.

But you know these are these are kind of old school methods that we used to use you know in earlier days, but you know these days we basically use more sophisticated techniques. So, in the remaining part of the this chapter we will discuss even more sophisticated techniques, ok.

Thank you.