SIX DEGREES OF SEPERATION: MEET YOUR FAVOURITES 03

So as i had said i have downloaded the facebook data see edge list format as i had explained as been used here there is an edge between person zero and person one person zero and person two and so on see the persons have been anonymous that is if you don't know the name and other details about the person they have been associated some dummy id and that id is used to identify the persons that is the way you identify the nodes in the network so it's an anonymous data you can download from the link that i had shown in the power point and it will be available in the description as well you can check out that and you can download this data. Now given that we have seen our data let us get to programming alright to deal with graphs we have package in python called networkx let me import it import network and given that the name is linked here let me use a shorter alias name as n xi will call nx so this is how the package that you used to analyse the graphs so the as i had shown the data was in edge list format so i had to read that data so for that the functionality is nx dot read underscore edge list see the names are also intitutive you can understand what it does and see you are getting the arguments you have to give the path of the file in my case the file is in the same path as that of my programme so i am just giving the name of the file, if your file is in different path you need to specify the complete path. So the name of the file is facebook underscore combine dot txt that is the name of the file alright may be i will maximise this pane so that its easier for you so i have maximised this pane so you have read the edge list so as a result of reading the edge list a graph object would be created that is sort of graph structure would be created and we need to capture the created structure so we will capture it by the variable G, G for graph basically so will capture it this way alright so what i am going to do is, i am going to compare the shortest path length by length i mean the number of edges you have to pass in a if you take the shortest path, that is in our example from Chennai to Kolkata, Kolkata to guwahati was the shortest path that is there we are two edges Chennai to Kolkata and Kolkata to guwahati so that two is the length of the shortest path is what i say. So like that i am going to find the length of each pair of nodes the shortest path that is the shortest path of the each pair of nodes, is what i am going to find. So for that i need to take the list of all the nodes right? let me capture that in a variable N, N for nodes basically let me say is equal to list of the nodes is what i want so let me type cast to the list and i will append the nodes of the graphs by telling G dot nodes this is how you get the nodes of the graphs and you are converting it into a list format so you are getting the list of the nodes, so that is what you are doing as well as the intention that i am having here is i am going to find the length of the shortest path between each pair of nodes each pair of nodes possible so between node number zero and one, zero and two, zero and three, one and two, one and three, one and four and so on so on each pair of nodes possible i am going to find length of the shortest path and i am going to observe something so let me do that i will for that i need to store the length of the shortest path in some list to observe something so let me store it, let me say shortest path list shortest path list so shortest path length list let me say i need a list and i am having an empty list here and now as i said i am going to check for each pair of nodes right so i need to run a loop and iterate over all these possible nodes so i will run a loop for u, u is a node let us say for u in nodes we have capture the list of nodes in the variable N u in N as well as we need tyres like this u would capture one of the node we need to capture another node so that we find the length of the shortest path between that tyre so we will find another will take another node for v in N so what i am going to do is i am going to find the length of the shortest path but it should not be found for all the nodes because if both the nodes are the same then it doesn’t make sense right i want to go from Kolkata to Kolkata it doesn’t make sense right so these two nodes must be different so only if they are different you should go for finding the shortest path length so i am going to give the check here u s not equal to v that is the nodes are different in that case you have to find the length of the shortest path let me call it l for length so for that we have a functionality in networkx nx dot shortest path length see you have a functionality here shortest path length G source target this is the format so you have to pass the graph object G and your source node is u target node is v between these two nodes you want to find the length of the shortest path and let us print it print let us say shortest path between u and v is of length l we have captured the length on variable l so will display that l here as well as i had said i am going to append each of the shortest path calculator onto this particular list shortest path length list so i will append that shortest path length list dot append the value of l so till now what i have done? I have read the edge list format graph data and i have captured it into a graph object first i had imported the network nx library then i had used it to read the edge list format data and i have imported it into a graph object then i capture the list of all the nodes from the edge list that is the edges where given from that we have to capture the list of all the nodes that we had done and we are planning to find the shortest path length between all possible pair of nodes and do some analysis on it. That is our aim for that will have list, shortest path length list which will store the list of all the shortest path lengths, all possible pair’s shortest path length it will be stored here and so we need to find the length of all possible pair shortest path so that for that we are doing we are iterating over the list of nodes we take a node u we take another node v if they both are different we go for computing the shortest path length and we print the length of the shortest path and we append it to our list, so by the end of this loop all pair shortest path is calculated and you have that stored in this particular list. So now from this i want do find the minimum value that is the least value of the shortest path between the two pair of nodes that i can say i can say the minimum value of the shortest path let me say minimum shortest path length is nothing but i want to find the minimum value from this list, this list will contain the lengths of all pairs shortest path so i want to find the minimum value so let me find the minimum shortest path length list from this list i want to find the minimum value just like minimum i want to find the maximum let me copy and paste it and modify i will copy i will paste this i will say maximum so for this i have to use the functionality max so min is the functionality that will return the minimum element in a list max is a functionality that will return the maximum element in the list so maximum of the all the shortest path length would be returned here and the minimum value would be returned here so given that we have found the minimum and maximum what next mathematically? Average right? you want to find the average shortest path length so to find the average we need package called numpy let us import it import numpy this particular package has most of the mathematical operators that we generally use average operator is also defined in this particular package so i have imported numpy and let me find the average so average shortest path length is nothing but in numpy package you have average function see you could get it here average and from this list shortest path length list from this list you want to find the average alright you have found it now the next job is you want to print all these values so let us print minimum shortest path length ok that is stored in the variable minimum shortest path length alright just like this we want to print the maximum as well as average that whatever we have computed we want to print the values so let me copy, copy and paste here and let me make the modification, minimum and maximum and next we want is maximum and that particular thing is stored in the variable max spl so we will have it here and the next thing is we want to find the average. Average shortest path length average shortest path length alright so we have done we have done with our program so maybe we will just i will save it and we will have a quick overview to analyse the graph data set we need networkx package we have imported it and see we are reading it from the edge list format we have downloaded we read the edge list format and we captured the graph object and from that graph we are extracting the list of all the nodes and then we want to find the shortest path length between each possible pair of nodes in the graph. So that is our intension for that to store all the lengths and analyse later we are having a list which stores all possible shortest path length. So to find shortest path length between all possible pair of the nodes we have loops two for loops running you pick a node u, you pick another node v and if they are different you find their shortest path print it and append it to the list that is what you are doing so at the end of this loops what happens is between every possible pair of nodes whatever the length of the shortest path it would be stored in this particular list. So now we are analysing this list, we are finding the minimum of all possible values that is all the values of all shortest path lengths. We are finding the minimum then maximum then the average to find average we need library called numpy so we have imported numpy in numpy the functionality of average has been defined and we are importing it. So given that we have computed all these values we will be printing it that is what we are doing and by six degrees of separations what we mean is the average shortest path length on an average you can reach from anyone to anyone on the earth if you take the shortest path by maximum six that is the average shortest path length would be maximum six this is what we mean by six degrees of separations, so let me give you an intuition for why this is true. So just assume that you have hundred friends in your facebook so we generally tend to make friends with people who are like us right so those friends may also have further hundred friends so the next level if you consider your friends friends that is hundred into hundred and the next level hundred into hundred into hundred see if you go just one level below the number of people you met is increasing exponentially so if you trace just sic levels you can definitely cover the entire population of the world. This is the intuition behind six degrees of separation concept ok i hope you are clear with the concept, you are clear with the terminologies of graph data structure how to deal with it, how to code it, i hope you are clear with it. Maybe you can pause here think for a while and you can then proceed with running the code and this is not just only functionality available here actually there are a lot of functionalities available here maybe let me show you. Nx dot i will just press a tab sorry i am sorry nx dot nx i will show you the other functionalities nx dot see a lot of functionalities available here and you can go to the console as you know by typing in the functionality and giving the question mark at the end you would get the documentation of what that functionality is all about. If you have some idea of graph theory the math behind it, if you have some idea you can understand it really well but still even if you don’t have it it’s never a problem all this is nothing but the intuitive thing translated into hardcore mathematics that’s it its really very easy you can pick up that so you can i would recommend that you guys explore this networkx library really well you understand the various functionalities, you see that what all can be done you will realise that it has whole lot of things which is otherwise difficult if this is not present. If you are asked to do this all pairs shortest path to find the shortest path there is an algorithm that runs for some the mathematics behind it everything it runs for some pages in some standard text books if you see. so we have this particular functionalities that finds the shortest paths that makes our life easier if we have to implement this from the scratch it would take a lot of time. So networkx really simplifies a tasks of dealing with graph so you can analyse the parameters of the graph without having to worry about the minute details that is advantage of using networkx. Alright please do explore it ok let me save and run it so before that let me restore the pane alright so here we have the output will be getting the output in the console so let me run the file. See the shortest path is being computed between the different pair of nodes. Are you getting somewhere more than six the average will convert to six it will take a lot of time because see there are so many nodes see three thousand nine hundred some nodes are there so there are around four thousand nodes i guess, you can check out that in the description of the data set as well, how many nodes and How many edges are present is given there, even that there are so many nodes present it would take some time to compute but you can see what is the minimum maximum average also do explore the other parameters and please do discuss in the discussion form of your explorations. Thanks for watching this video, have a nice day.