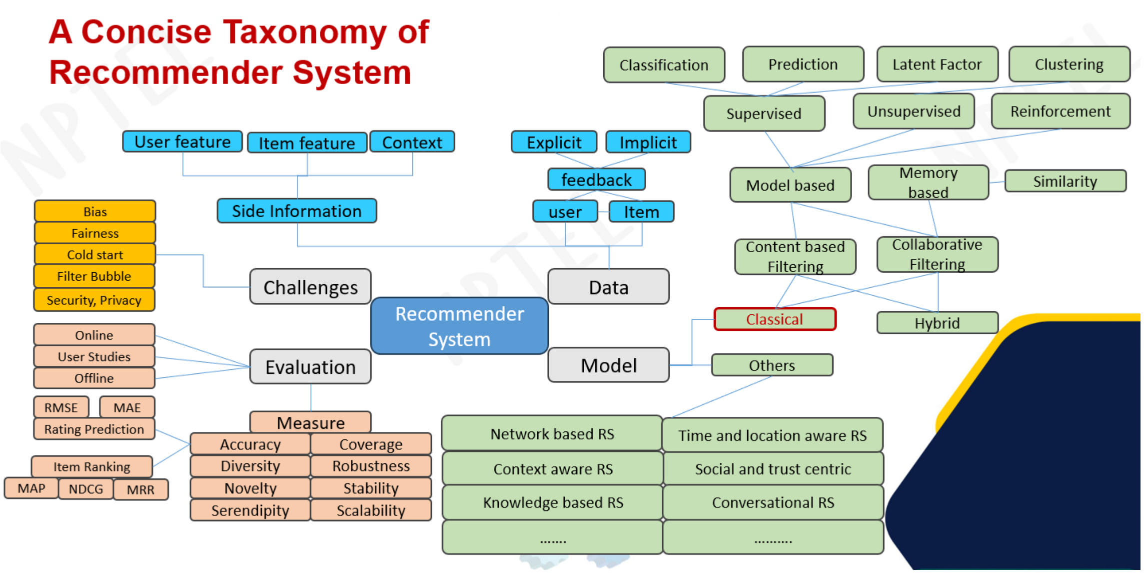
Course Name - Recommender Systems
Professor Name - Prof. Mamata Jenamani
Department Name - Industrial and Systems Engineering
Institute Name - Indian Institute of Technology Kharagpur
Week - 01
Lecture - 02

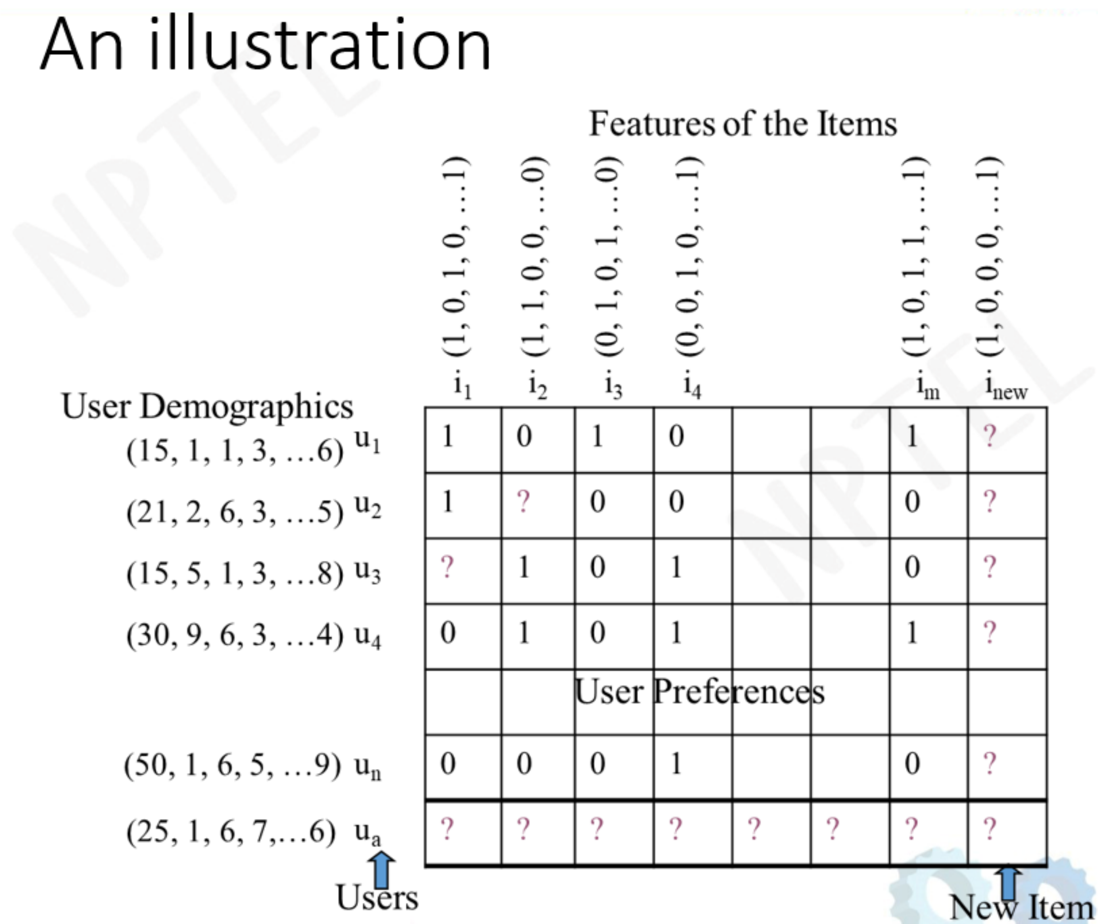
Lecture 2: Types of Recommender System-I

Hello everyone. Today is the second lecture and it is again the introductory part. Today we are going to talk about Types of Recommender System. Moving ahead in this we are going to cover first a framework for understanding recommender system, then based on that we will be talking about different types of recommender system. So, ah as I have told you in my intro slide, ah we have four aspects while studying about recommender system. Model is of course, the most important aspect, but besides that we are also going to talk about the data which is responsible based on which you will be running your recommendation algorithms.



So, to start with let us just have look at this data. Here we can see the major data source is the user rating which comes from the users rating the items. This can be again based on directly asking the users that is explicit otherwise it can be implicit. There can be as many side information as well.

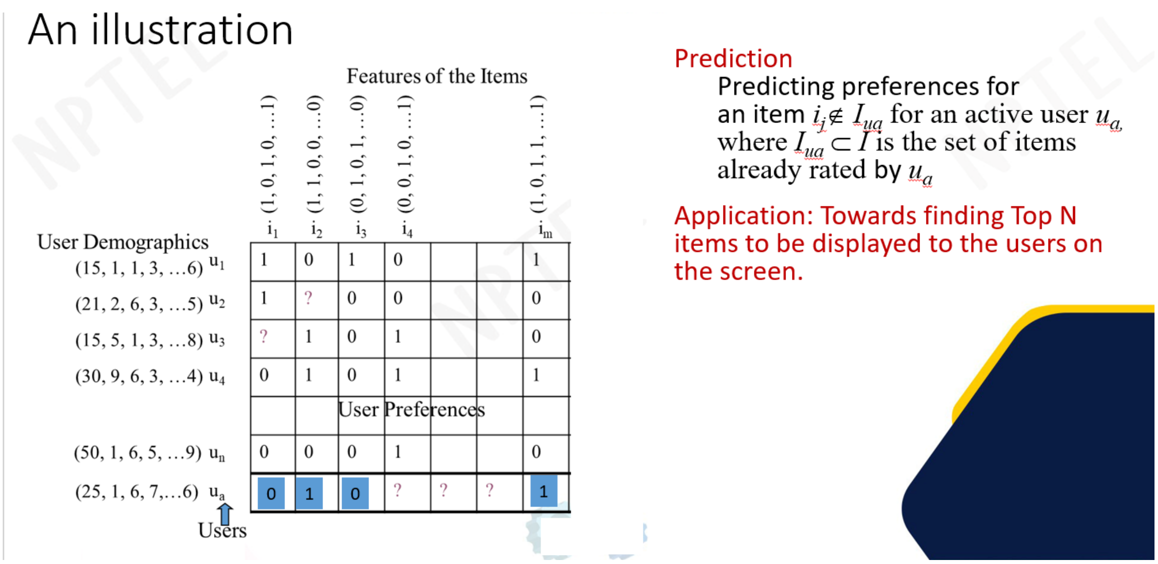
So, in a typical recommender system there are three major components. First one is the user, second one is the set of items and third one is the ratings which are given by the users to the items. So, this user data is basically the user demographics data, then these are the features of the item and in this matrix we have user preferences. The set of users we represented by this vector u which consists of m elements, m number of users, n number of items and this this rating matrix R consequently is a m cross n matrix. This is one illustration.



This user u which we said is a set of users u 1 to u m each u is a vector. So, each u let us say u i consists of many elements. What are those elements? Those are individual user features. So, this could be let us say the age of the user, then something let us say his gender of the user or his liking about certain items and many more such elements. So, this vector when we consider those vectors together they make one matrix.

Similarly, we have the items. Here each item i is represented by a set of 0s and 1s which indicates presence or absence of certain feature and this again makes one matrix. So, this item let us say i j consists of certain 0s, 1s, 0s, 1s, 0s and so on. Now, let me tell you when we talk about this user features and item features in this illustration we have used here certain numeric values and here only 0s and 1s, but in a typical situation this may not be so you may have in case of items you may have actual some kind of numeric values. Here we indicate simply certain presence or absence of features, but you can have let us say for example, if we consider news items.

In news item when we talk about this how to extract this text features to represent news item we will know about there is something called TF-IDF rating which you will be filling up here. The TF-IDF rating will be some kind of real number. So, it need not be always what I mean to say this need not always be a 0 1 vector. Similarly, here also it need not be the exactly whatever I am showing, but this will contain certain values. Now, come to the third matrix this is the third matrix third matrix is a m cross n matrix. So, you have some m number of users and certain I just made the opposite n number of users and m number of items. So, this matrix is again a n cross m matrix. So, here again the entries we have shown as 1 and 0's. One indicating whether the person has given certain rating as 1 and 0 maybe he has not given n rating. Now, this 0 is again a very tricky situation when you we consider it as rating matrix. 0 means either absence of the rating or it may indicate that the rating is not given by the user. So, in case it indicates that user has given a negative rating then probably this 0 will be indicating the opposite of 1. Otherwise, if you we mean that the user is not willing to give any rating then this indicate this 0 indicates absence of the rating. So, it can be a even if it is a binary variable this becomes an asymmetric binary variable ok. So, now, again in when we talk about this rating matrix in this particular case we have shown only 0's and 1's.



	i_1 (1, 0, 1, 0, …1)	i_2 (1, 1, 0, 0, …0)	i_3 (0, 1, 0, 1, …0)	i_4 (0, 0, 1, 0, …1)			i_m (1, 0, 1, 1, …1)
u_1	1	0	1	0			1
u_2	1	?	0	0			0
u_3	?	1	0	1			0
u_4	0	1	0	1			1
u_n	0	0	0	1			0
u_a	0	1	0	?	?	?	1

Now, when we consider this 0's and 1 in a typical setting like that of let us say u tube we say u means 1 means something is present and 0 means something is absent. So, you must have seen there is a thumbs up and thumbs down. Thumbs up means you are appreciating the item, thumbs down means you are not appreciating the item. Now, one more observation about this matrix. You see in this matrix many places like this this one, this one, this one and so on there are question marks.

So, what are these question marks? Here this question marks means in these places the user has not given any rating. Now, why user is not giving any rating? In a typical recommender system setting in a consider a general store online store from where you are trying to make some purchase. The store has millions of item out of this how many you have actually seen, how many you have actually rated very few. So, therefore, many places if you are this let us say user 3 in many places wherever you have not given any rating there will be this question mark. So, these blank entries are something which needs to be estimated in the process of a running recommendation algorithm.

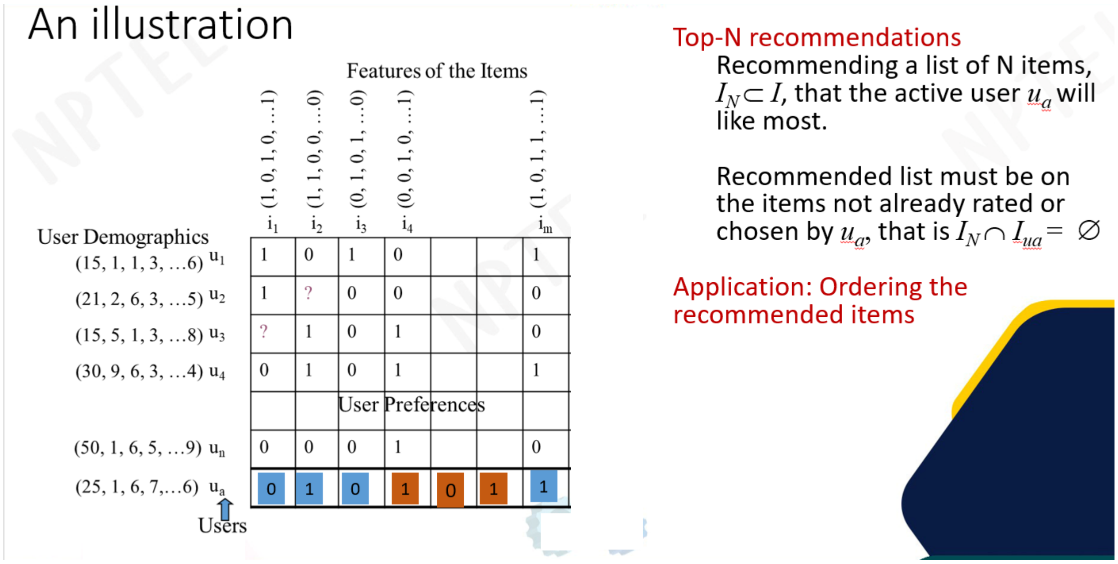
Now, let us look at what are the typical decision making scenarios in a recommender system. By decision making scenario we mean what kind of outcome will come out of the recommendation algorithm. The first task is we have to make the prediction. Prediction of what? Prediction of the rating maybe and if we predict the rating which are the items for which we have to predict the rating. The area where we have this question marks. Then we have top n recommendations. What is this top n recommendations? We have whatever recommendations we have whatever values we have predicted those values we have to now sort and show to the user. Then we have top m user recommendation as well. So, as we move ahead we look little bit in detail.

Now look at this. This is a prediction scenario. In this prediction scenario this is the user A for whom we are trying to predict the rating. Now this user A has already given ratings to few of the items. So, based on this we have to give the ratings for this question mark field. So, how exactly we do it? There are various ways to do it.

One of the way could be we have to find out the like minded users who have already given the ratings to the items which user has given. For example, this one user 4 has given in 4 places of course, here it has not given. So, now, this there is another user let us say this one this has given in 2 places like him. So, similarly the looking at this rating matrix only we can find out various places where other users have given rating to same items. So, as a result we can run certain procedure certain algorithms.

So, that based on whatever ratings they have given to the items which is not rated by this user u A which we call as the active user u A we can predict this value. So, now, once we predict these values where the recommendation is not given next comes the question how to display these recommendations to the user. So, now, you know that what are the predicted values for all this. Let us say predicted values for all this are of course, here we have given 1 and 0, but it can be continuous as well. Suppose this predicted values are 1, 0 and 1.

Now, which items will be shown to the user in a typical recommendation setting on a screen you are supposed to show this list these items. For example, if you are watching let us say Netflix movie the movies are to be sorted based on the kind of prediction that has happened. Of course, here we have 1, 0 and 1 which means these items are supposed to be liked by the user. So, therefore, you will be recommending these 2. Suppose this would have been certain continuous value like let us say this would have been 0.5 and this would have been let us say 0.2 assuming that this whole matrix also has the values between 0 to 1 this is 0.2 and this is 0.1 maybe. In that case the sorted value would have been first you would have shown item 4, then next option would have been this item 5 and finally, if there is any space on this screen then you would have shown this item.



So, which means now you have to sort this item based on this predicted values or you have to find out certain strategy. For example, here we have 1 for this suppose there are certain 100 number of 1's in the screen you do not have those 100 spaces. So, you have to choose somehow the top few which are supposed to be shown to the user. So, the idea here is you have to recommend you have to recommend a list of n items where n is a subset of i that the active user ua will like most which means you are trying to predict which are the items the user will like most. So, in this case these are the items which the user is going to like most this is 1 first 1 this is second 1 this is not going to be like if we consider this binary scale and if we consider let us say this would have been a numeric scale then this 0.5 we would have sorted it in the order 0.5, 0.2 and 0.1.

Now next is your top m users suppose a new item i1 has come to the store. Now based on some algorithm we have somehow estimated what are the predicted rating of this item for these people this u1 to un. Once we have these predictions ready then we can target the users who are likely to like this item who are likely to order this item. So, therefore, to them we will try to show this. So, this basically can be used some kind of making some kind of target advertisement. So, the setting here is little bit different now probably the algorithm that was worked fine for the last case where we were trying to find out the top m top n number of items now the algorithm probably will be different.

Now probably we have to figure out which items are similar to this item and based on that similarity score probably we will have to make this prediction. So, therefore, the algorithms will change as the nature of the decision making situation changes. So, this is one example of top n recommendation this example you have already seen in some places I have already shown you here we are trying to buy this recommender system handbook and it shows what are the other items which are bought together along with this item. Now look at this there are three number of pages out of this three number of pages first page will be first displayed. So, how do they decide this order? Ok. So, deciding this order is top n recommendation similarly deciding the order which users are going to like this out item is your top m recommendation.

Moving ahead now we can formally define the recommender system. A recommender system can be abstracted to be consisting of a user model, a community, an item model and a recommendation algorithm and an interaction style. Now what is users model? Let us say in the last example that we are talking about we were trying to based on the other users rating we are trying to predict the rating for a specific item which is not already rated by this user. So, in that case we made our user model based on certain kind of some kind of similarity score with other users who have already rated this item ok.

Now what is the community? Community is those set of users based on whom we have taken the decision. So, this is our community model then product model look at the second situation where we were trying to find out the top m recommendation. In that case we used the item features we made a model for the item and based on the probably on the item similarity we tried predicting the rating. So, we have a product model. So, product model again we have seen that in that matrix each product will have a number of features and all the products together if we put together we make one item item feature matrix where the rows will be individual items and columns will be features or the columns will be features and rows will be individual items.

Now then comes the recommendation algorithm. As we saw in every cases the situation is different. So, we have to use different types of recommendation algorithm where we can reason out how to predict this ratings. Then comes the interaction strategy. This let me tell you this interaction strategy is something which probably we are not going to discuss. So, this is all about how the recommender system will be interacting with the users. So, how the recommender system will interact with the user? You user will be clicking some button then user has to select something user has to view something. So, how are those user interaction are to be organized so that user gets maximum exposures to this recommended items and take takes certain decision. So, this can be done through designing certain good recommendation strategy. So, this is the what I already said these are using this user demographic probably we can make one user model.

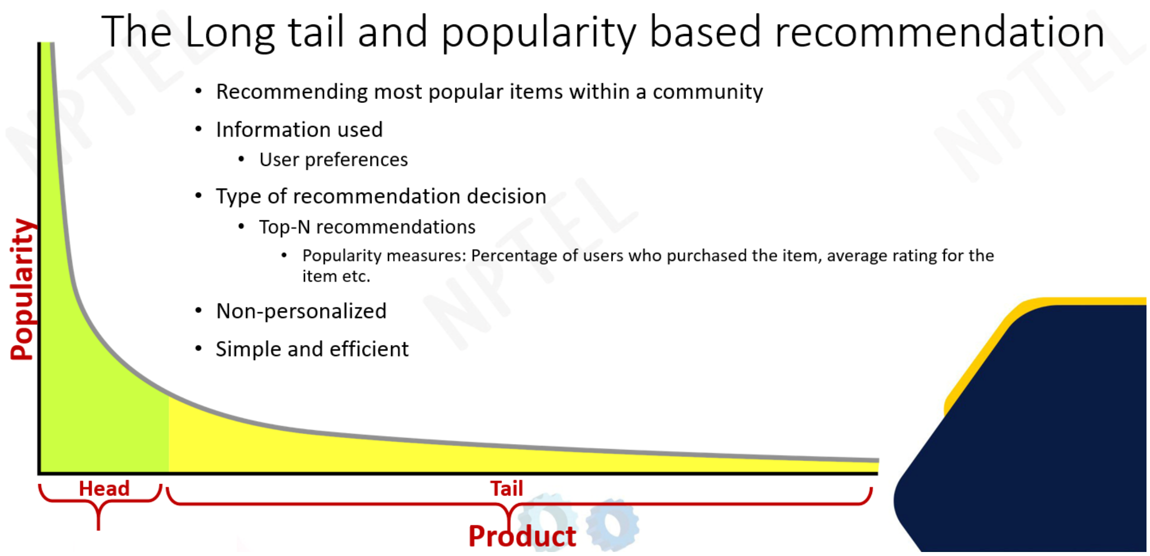
Similarly using this similarity among these users we can make one user model from this community. These are the product features we can make a product model out of it. Then we can have recommendation algorithm which will be deciding this question in marked places where you will be predicting certain values. Then you can have interaction strategy where you have to decide what should be your choice of words, what should be your choice of visuals and at in what kind of physical space the user is going to use this interaction. For example, if you are showing this on a let us say mobile device let us say same page on a mobile device probably have a different look than if you are using a laptop. So, in a mobile device it has to be managed in a manner so that even understanding position people can take some kind of actions without much difficulty which time you are giving this recommendation. Then what are the possible behavior you expect if you give this recommendation. So, based on all this you can decide your user interaction strategy which anyway we are not going to cover in this course. Now, look at we are we have again come back to the same figure. Now, we have come back to the we will be in fact, coming back to this same figure for different reason.

So, now right now after knowing the basic data arrangement in a recommender system the basic type of data that you can come across in a recommender system we are going to have a brief overview on different kinds of algorithms. See on this data we are going to look little bit in detail at a later stage just now I give you a overview of the data. So, similarly now I am going to give a overview of the models. If you look at these models there are certain classical models which are widely discussed from the very inception of this idea of recommender system. The content based collaborative filtering and content based filtering adds two such mechanisms.

In their purest form they have certain ways in which you can make this rating prediction there are certain hybrids as well which combines the idea of both. Now, there are various ways you can make models and all, but those things we are going to deal in detail later on. Right now let us just look at what are various types of recommendation systems. So, if we look at various types of recommendation system as I told you these are two basic types collaborative filtering and content based. Besides that, there is one recommender system which you might have seen very frequently which is popularity based. Most of the e-commerce sites or streaming services will be giving you what are the popular items which are now hot in the market. So, these are basically non-personalized recommendations, but these two are personalized recommendations. So, also the others. So, as we move ahead we see the kind of data that just now we saw are typically used here, but as we move ahead we get more side information like in case of reputation based we have to make something called some reputation matrix and use it along with our rating matrix.

So, also the case of trust based. In case of knowledge based we have to use certain additional domain knowledge. In case of social network we need to know how the users are connected with each other. In case of context based certain additional context such as time, location of the user we have to consider and we can also have hybrid of all this. In fact, this is not an exhaustive list as we you can go through the literature of course, this course is going to be a very basic course as you move ahead learning the ideas from here then you can see in the literature every day some new type of recommendation algorithm is coming up which caters to different voids that exist in the research and application area.

So, first one popularity based recommendation. So, here again one of our old slide I am trying to show. This is that long tail phenomena in which popular items are here and less popular items are towards the tail end. Now, if we look at this, this part which that time I told that these are the popular items and they are typically stored in a physical store, but in a retail store rest of the items can also be available at least in the database. Now, besides the physical store why these items are becoming popular because many people are viewing this. So, because many people are viewing this, these are some of the at that particular point of time these are some of the very relevant items.



So, therefore, it is very likely that other users will be interested. So, this is the most simple one if you look at let us say your streaming site like that of Netflix you will find out what are the recent items which are popular, popular movies, popular documentary, popular news items and so on. You will find out on your probably on the that particular streaming site. So, here user preferences as I told you this user preference can be directly collected or maybe indirectly in an implicit way it can be collected. So, more people watch this item more popular they become. Sometimes even you say you can you must have seen some of the items which are typically not very recent they also become popular. So, it is not about the recency it is about which items are going to be popular in that particular time. So, this popularity measure can be of various types percentage of user who purchase the items, average rating of the items and so on. But it typically depends on the kind of application in which, but for sure once you have this popularity measures you can items based on this measure and you can provide the top n items. As I know as I already told you this top n item list is actually a one very large list only depending on the screen size on which the user is viewing the recommendation you can fit it. So, you must have marked depending on let us say your mobile size is very small you will be getting probably top 2 recommendation. If your laptop size is large you will be getting large number of recommendation, but it whatever may be the case if it is a popularity based recommendation whatever you are getting the same thing I am also going to get. So, it is a non-personalized recommendation ok. So, this is a simple one. So, I think we will be now ending this lecture here and we move ahead.

So, in our next lecture other kind of recommender systems we will be introducing these are some of the references I used and I will also be using the same references in the next lecture. Here we tried looking at the major data sources that are used by classical recommender system. There are three major data sources one is user detail, then item detail and then preferences. And this preferences is in the form of a matrix which has the dimension the number of user cross number of items. Similarly, there are three major decision making processes here prediction of user preference, top n reference preferences and top m number of users. When I say top n number of preferences it is top number of preferences for the items. So, with this I end this lecture. Thank you everyone.