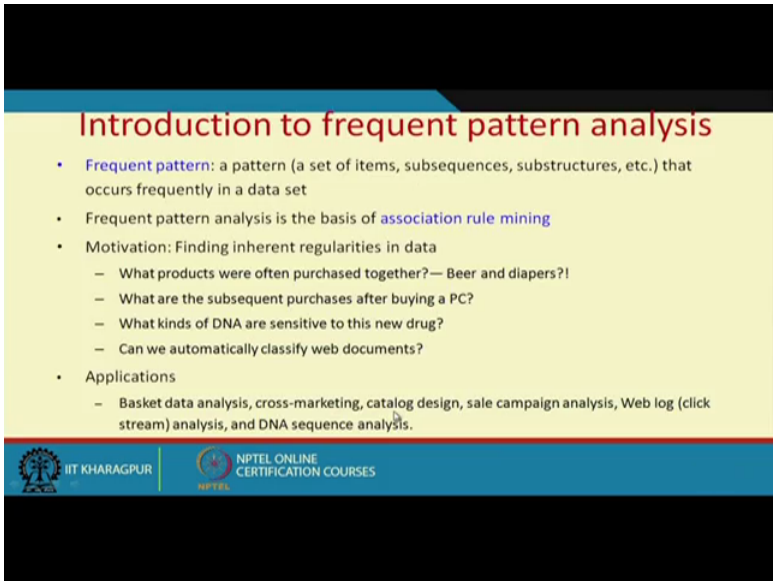
E Business
Professor Mamata Jenamani
Department of Industrial and Systems Engineering
Indian Institute of Technology Kharagpur
Lecture 55
Association and Demographics Based Recommender System

Welcome back, today we will continue our discussion on recommender system. So far in this particular series of lectures talked on recommender system we have seen the basic concept of recommender system and different types of recommendation system then we have talked about content based and collaborative filtering. In fact, some are very simple and we are not going to discuss about them and some very complex theories related to recommender system also we are not discussing. What we are doing, the concept of recommender system we are explaining with the help of few examples in this series. Next class that is today's class we are going to discuss on Association based recommender system.

(Refer Slide Time: 1:12)

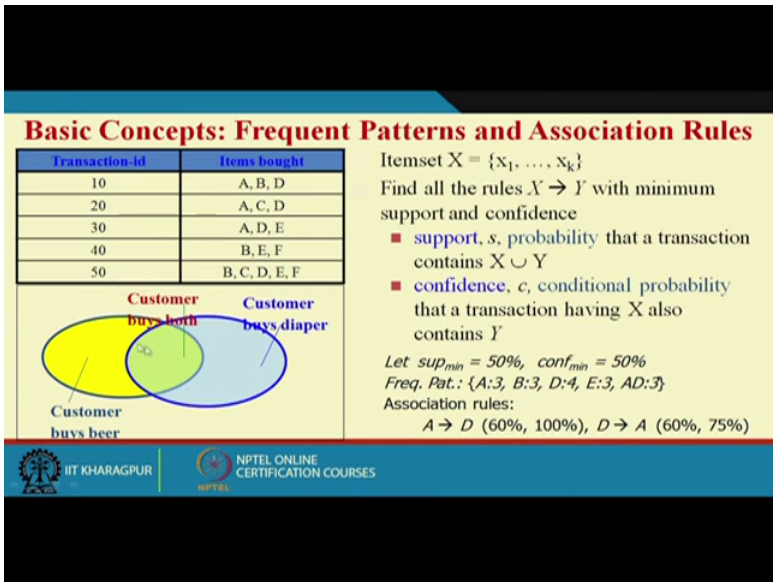


So in this Association based recommender system, the basic idea on which this Association based recommender system is based on is Association rule mining. In fact, while talking about talking on the analysis of access log we had little bit discussion on this particular topic like how Association rule mining can be used to connect to find out which pages are browsed together within a session and what are those frequent sequence of pages which are browsed within itself each session. So now today we are going to again see how this particular methodology and be applied in recommender system in recommender system context while recommending items to the users.

Here let me once again remind you, while discussing about the concept of recommender system we know that recommender system basically while the data is used for recommendation is of 3 types; one is user item rating matrix, one is user demographic data, another is item details, okay so let us try to understand that how this data can be used for generating recommendation using Association rule mining. The basis of association rule mining is your frequent pattern analysis, now what is the frequent pattern? A frequent pattern is a set of items, subsequence or substructures that occurs frequently in a dataset. So in this context that class we discussed about the example of market analysis where the study is about to find out which items are purchased together.

So that market basket analysis is most frequently cited example of this Association rule mining. However, Association rule mining is widely applied as we have seen in case of access log analysis; here we are going to sit in the context of recommender system. Now let us see what is this frequent pattern analysis, okay.

(Refer Slide Time: 4:00)



So this is the example that we studied like if we have 2 sets of customers, the customer who buy a particular item, the second set of customers who buy another set of items, the intersection of these 2 sets is a set of customers who buy both okay. So let us say we have many transactions happening where the items bought or items rated together are given here, so which means you can associate in the context of recommender system you can associate it with the idea of many items viewed together in a particular session or many items purchased together in a particular session. In the context of e commerce, when you visit e commerce site many times you view items together but you do not purchase.

So irrespective of if you would like to suggest the items, it is not always that items need to be bought together, many times what the users do? They will be looking at the items in a online store but you will be buying the item in the physical store, so therefore for the purpose of collecting the data even when the items are viewed together and collected okay. So now each user has viewed a number of items or maybe he has purchased a number of items together in a particular session, let each transaction be that so these are the transaction IDs and these are the items which are bought or viewed together. Now each of these items which is say is one item set, now within the set of items our aim is to find out all the rules all the rules where X represents the set of items X represents set of items when they are bought, Y is also what.

Now if you consider these rules, many rules will be generated for example, here itself we can say when A is, we have the evidence that when A is bought, B and D are also bought. When A, B are bought, D is also bought. So from each of these transactions you can generate many rules. However, if you consider these rules some of these are most frequently occurred, now in order to identify which are the rules which occur most frequently many statistics are used, many matrix are used so 2 such important matrix are support and confidence, what is support? Support is the probability that a transaction contains both X and Y. Now competency is the conditional probability that a transaction having X also contains Y.

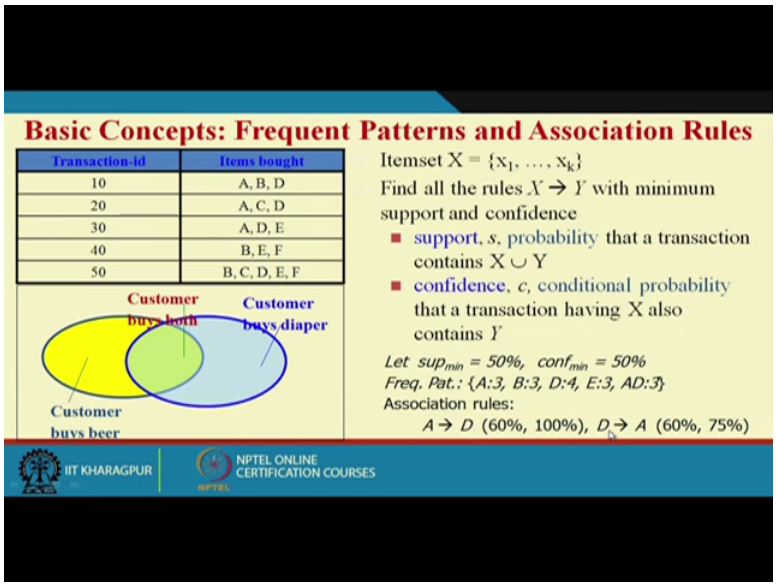
So look at this example, assume that you now see this minimum support and minimum confidence, you can decide yourself from your dataset I mean depending on your dataset. Now suppose we assume that minimum support is 50 percent and minimum confidence is also 50 percent, and suppose we got the frequent pattern that A has occurred 3 times, look A has occurred 3 times, B has also occurred 3 times, D had occurred 4 times, E has occurred 1, 2, 3 times, F has occurred how many times? F has occurred twice.

Now what is my minimum support, my minimum support is 50 percent so how whether I should consider A format no, I cannot consider because I have total 5 transactions, 10, 20, these are transaction IDs, I have total 5 transactions out of which half should contain, half should have the evidence of occurrence of a specific group of items or a specific item. So in that case if you consider single items individually, only A, B, D and E are satisfying this. and what about C? C has occurred twice which is less than 2.5 and F has occurred twice which is also less than 3 less than 2.5. And coming to 2 item pairs A, B is one pair, A, C is another pair and so on, now what is the evidence of A, B occurring together, A, B has occurred only once.

B, C has occurred only once, but in case of out of these 2 item pairs only A, D has occurred 3 times. Now consider 3 item pair, is there any 3 item pair which is occurring more than 2.5 times I mean 2.5 cannot occur, it is 3 anything occurring 3 times? No, only A, D has occurred 3 times. Now, when these are my frequently occurred items item set, right now this is A, B and D cannot be considered because they are single item sets. I can only find out the Association rule of more than one item occurring together from this A, D rule itself so my association rule is, when there is A in the transaction, A is seen by the buyer, A is bought by the buyer then D is also purchased, when these purchased by the buyer A is also purchased.

Now with minimum support I have found out these 2 rules okay. Now this first number in this bracket represents the support, so I have 60 percent support, 60 percent of this that is 60 percent of this 5 transactions is 3 transactions so 3 transactions I have the evidence of 3 transactions supporting my this rule, now let us try to find out what is my confidence. Now what is the difference in the confidence? It is the conditional probability that a transaction having X also contains Y, which means is A is there, D is also there. Now look, when A is there D is there, A is there D is there here A is there D is also there, so I have 100 percent confidence that if A is put in the basket, A is purchased, A is viewed then D is also purchased, D is also viewed whatever may be the case.

(Refer Slide Time: 11:53)

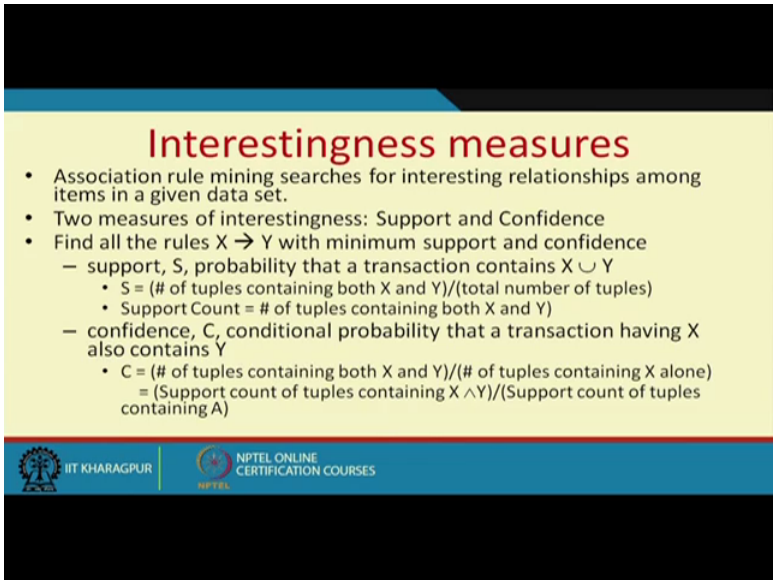


Now look at, if my rule is whenever D is there in the basket, A also appears in the basket. So in this context look at this transaction when D is there A is also there, D is there A is also there, D is there A is also there, in 3 transactions it is happening in this manner, but when you look at the fifth transaction, even if D is there A is not there so therefore out of 4 I have the

evidence of occurring D and A together only in 3 transactions. So my conditional probability of transaction having D containing A is 75 percent so here I have 100 percent confidence, here I have 75 percent confidence, but in both the cases anyway my confidence level is greater than 50 percent so therefore I will be going for these 2 rules okay.

So which means if the item A is seen by the buyer and I have from the previous transaction I have my association rules ready, I will be looking for the left side of the Association rule if the item purchased is there in the left side of the Association rule, I will be suggesting the item in the right side of the Association rule. Now if I have very large number of rules, what do I do? I choose the one with very high confidence okay. Now let us go little bit more details about it, this is what we have already discussed, let me just remind you once again.

(Refer Slide Time: 13:53)



Now this it about this interestingness measures, the support and confidence these are interestingness measures. In fact, when we talk about Association rule mining these 2 are not the only interestingness measures, there are many more measures as well but these are more these are widely used measures of interestingness and easy to compute. So Association rule mining searches for interesting relationship among items in the given data set, here we are considering 2 measures of interestingness, support and confidence and our aim is to find all the rules with their left side is X and right side is Y with minimum support and confidence.

Support S is the probability that a transaction contains both X and Y, I mean this is if you consider the entire dataset out of all the datasets where both the items have occurred together that is my support so the percentage of places where both the items are obtained together is

my support. My support count which instead of representing in terms of probability or in terms of percentage I can tell support count. If my data is I have 50 tuples then out of that if my support I decide my support count to be 50 percent then 50 transactions has to contain the rule, contain the frequent item from which I am going to derive the rule.

Now next is the confidence as I have told you it is the conditional probability that a transaction having X also contains Y, so how to compute it? It is the number of tuples containing both X and Y divided by the tuple containing X alone okay. So this is about finding the frequent patterns, but once we find out the I mean how do you exactly find out the frequent pattern and therefore you generate the rules. There are many algorithm for this, just for the example demonstration purpose we have to then this apriori algorithm but let me tell you apriori algorithm is not a very efficient algorithm, and why is not efficient algorithm that we are going to see shortly so let us try discussing about apriori algorithm.

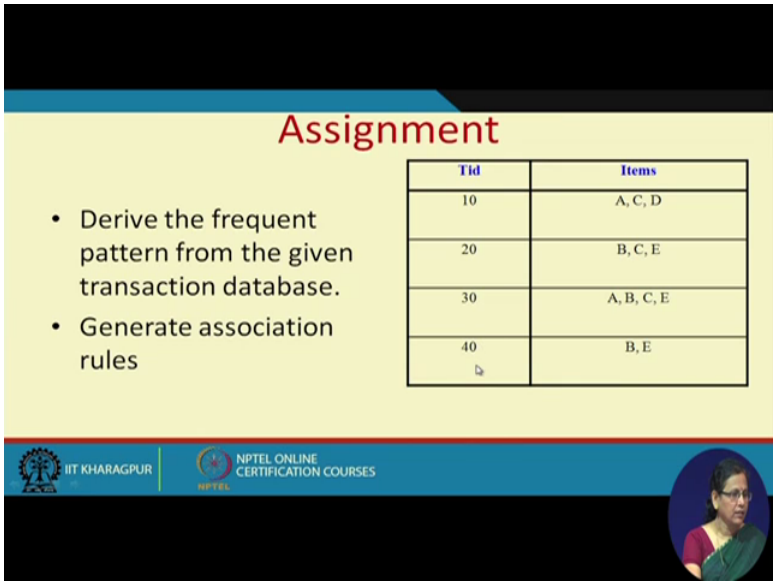
This apriori algorithm is built on apriori principle, now what is this apriori principle? This apriori principle says, suppose an item that is not frequent that is it does not have the minimum support then if you add another item to that set then the resulting set cannot be more frequent, which means like if we go back to our last example here one of the frequent pattern is D, D is in 4 places okay. So to this set if I add one more item it cannot occur more than 4 times, natural it will be a subset of that set, either the item will occur all the time for example, A, D has occurred 4 times but when we consider add another item to this set, they single term set D, if you add another item that is A, A is occurring only 3 times which means A and D are occurring 3 times, so it is anyway less than equal to 4 okay.

Similarly for A, A is occurring 3 times it is a frequent pattern. Now if I add another item let us say D it cannot be more than because A itself is occurring 3 times, how it can go beyond that? So it can occur maximum 3 times. Now so look if with A I add B, A is a frequent pattern 3 times occurring, B is another frequent pattern. Now A and B together will be either 3 times occurring or they will less time occurring, now look at this A, B is here in fact, we have only one evidence of A, B occurring together that is why A, B is not a frequent pattern in this case okay.

So this apriori principle which basically is an anti, monotone property, it says that if a set cannot pass a test then all its subsets will also fail the test. So if this apriori using this apriori principle this apriori algorithm is designed. So there are 2 steps in this apriori algorithm, one is Join and another is Prune. Now before we go through the details of this algorithm, let us

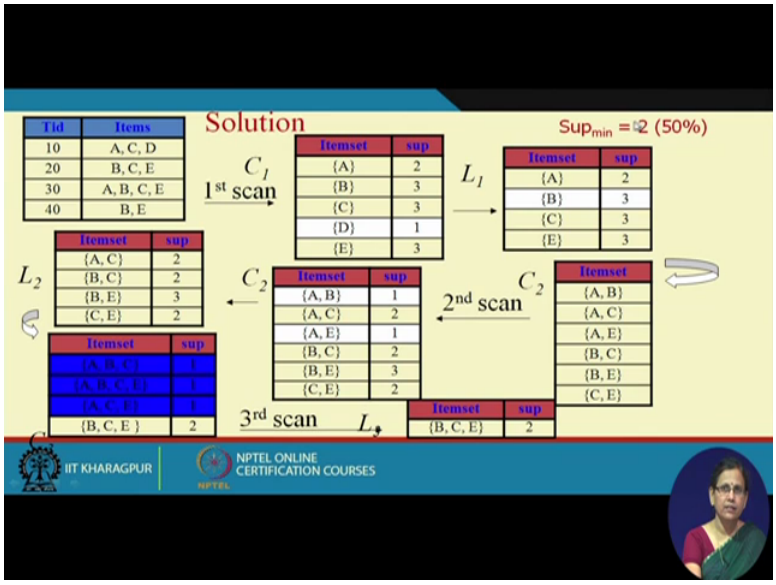
first find out through an example that the steps involved in this one then we go back here and go through the algorithm once again okay.

(Refer Slide Time: 19:50)



Now look, these are my transaction IDs, these are the items, let us try to generate association rules.

(Refer Slide Time: 20:02)



I have 4 transactions with all this, let my support count be 50 percent that is 2. So let us first find out what are the frequent patterns, so to start with I make a scan of the database containing the transactions and find out all the transactions with satisfying minimum support count. So the transactions first of all I will be starting with all the single term sets so A, B, C,

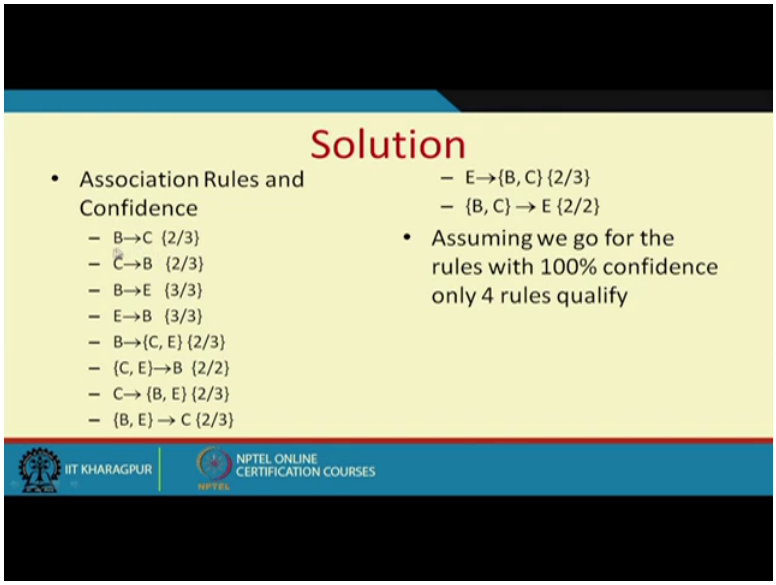
D, E together I mean individually how frequently they occur in this transaction set so this is 2, 3, 3, 1, etc. Now first task is as I have told you this algorithm has 2 steps; one is Join, another is Prune okay.

So now here first will be pruning, we will be removing that pattern which is not frequent, now I should remove this D which is occurring only once and not satisfying support count? The reason is since D is not a frequently occurred item, if I add one more item to this then that is not going to be frequently occurring so therefore I drop D and now I have A, B, C and E, these are the single terms which are occurring frequently. My next step is to join them together so I find out all the combinations of these 4, A can be combined with these 3, A, B, A, C, A, E, then B can be combined with these 2 B, C, B, E, then C can be combined with E so this makes all my 2 size 2 I mean the here the set size is 2.

So now my next task is I go back to the original dataset that is this transaction data set and search for the co, occurrence of these 2 items together, I mean occurrence of these sets. So what I do? I go back and in this transaction set, this set of transactions I search for this co, occurrence of these 2 items and find out its frequency. So A, B is occurring once, A, C is occurring once and twice, B, C I am taking this from these values then A, E is occurring once, B, C is occurring twice, B, E and B, E, B, E is occurring thrice and C, E is occurring twice. Now out of this again I validate this with my support count, my support count is 2 that is 50 percent of these transactions. So again following the priory principle I drop this one sorry I drop this one as well as this one.

After dropping these 2, I mean after pruning these to dropping means that pruning step, I am pruning these 2, I come up with these sets with 2 items each. Now next is again join, I combine these items sets so I combine A, C with B, C I get A, B, C, I combine A, C with B, E I got A, B, C, E, combine A, C with C, E I got A, C, E. Then I combine B, C with B, E I got B, C, E then B, C with C, E again that BC only then B, E and C, E is again B, C, E so I have these very combinations out of which again I am dropping this first 3 as they are not satisfying my minimum support count. So now I have these 3 items only one 3 item pair left so I do not have any further for joining it together okay.

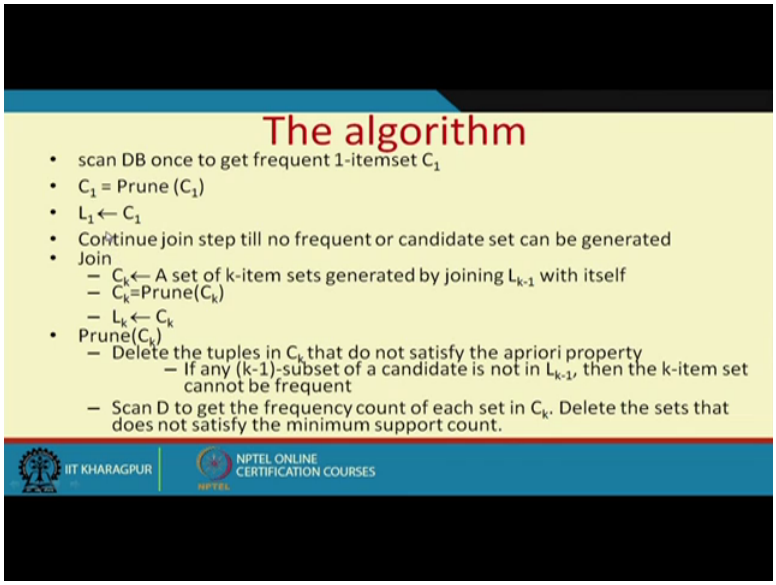
(Refer Slide Time: 25:15)



I have to now from this frequent items sets now I have to find out the rule, now what is my rule, how many rules I can generate? Look, this B, C, E, are co-occurring together so I have the evidence of B occurring when C is there, then C occurring when D is there, D occurring when E is there and so on, taking 1, 1, 1 items then taking left side one item and right side two items and so on. So after all this generated, next I have to check whether I have sufficient evidence of I mean how confidently I can say these rules I must accept, assume that I need to have 100 percent confidence. So now I again go back to the database and find out this confidence values, so to find out confidence value of B to C what do I do?

I go here B to C I mean when B is there, C is also there, how many times B has occurred? 1, 2, 3 and when B is there, C has occurred twice so out of 3, 2 transactions only I am confident with. So therefore what I will be doing, I will be writing my confidence as 2 by 3, so your task is to find out the confidence values of all this and if I assume my confidence to be 100% then how many such rules satisfies my confidence? Only few rules satisfy my confidence, B to E, E to B and C to E then I mean C, E to B and then B, C to E so these 4 rules now satisfy my confidence.

(Refer Slide Time: 27:22)



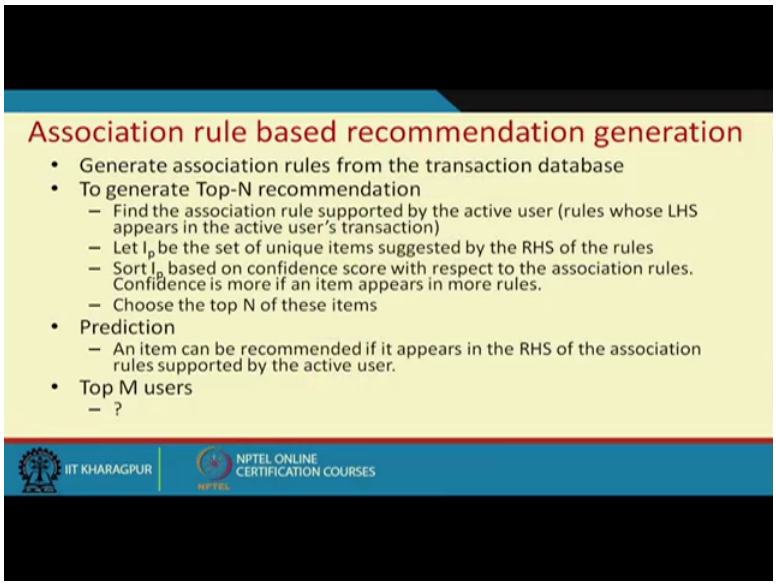
So from this association rule based recommender system in fact, as I promise you I must go back and show you the algorithm, this algorithm as I was telling you, why as I told you apriori algorithm we are just using to show you how we can suggest make recommendation based on association rule and association rule generation there are many algorithms of which I was just showing examples of frequent your apriori algorithm. But apriori algorithm as I told you is not very it requires a number of, it is not be efficient in the sense, it requires a number of transactions I mean a number of times you have to access the database, after each join while making prune once again we have to go back to the database and show how the elements I mean to find out the co-occurrence of the elements.

So here it is same thing is said here, first you scan the database to find out frequent one item set that is the first this one, only one element is there that you find out then First you execute prune then you put it in some temporary variable and you join it, the steps we have already discussed then you prune and you can dataset to get the frequency count of each. So this scanning, which each prune this scanning occurs so because every time you are accessing a secondary data secondary storage, here in the example we have shown only 4 items but in a typical transaction you think of Amazon, there will be millions of transactions.

If you consider purchase alone there will be tens of thousands, but if you look at the views how the items are viewed together that is what that is what they do that if the person who viewed this item also viewed this item, who search for this book also search for this book, which means co-occurrence is not only based on purchase, it is also based on how the items

were viewed together. So therefore, you can imagine how many rules will be generated so pruning these rules and every time going back to the database is a quite time-consuming process, but anyway this is one of the methods for recommendation generation that is what we studied.

(Refer Slide Time: 30:06)



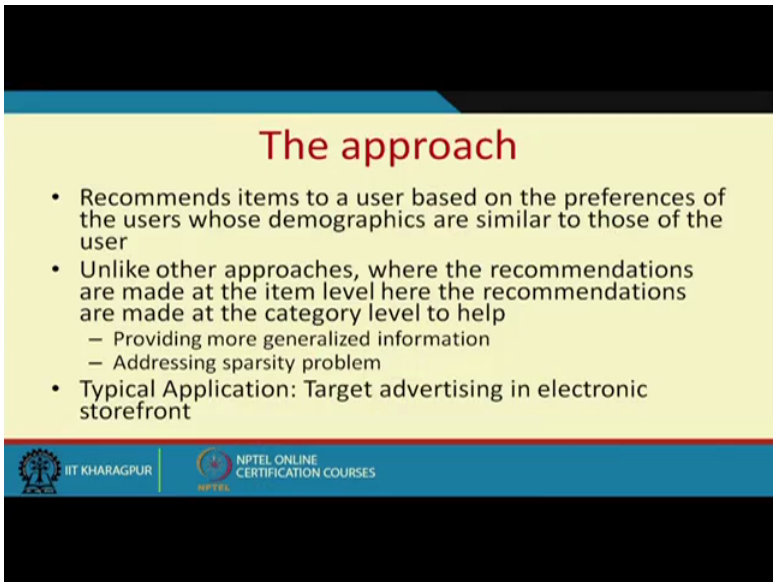
So this association rule mining generates association rules from the transaction database. Now to generate top-N recommendation what you have to do? You find the association rules supported by the active user whose left-hand side appears in the active user's transaction. Then let I p be the set of unique items suggested by the RHS of the rules, there will be many rules where the left-hand side will be the items chosen by the buyer and there will be one right-hand side which is another item. So what you will be doing? You will be you will be finding all the rules where the left-hand side is the items put in the basket or viewed together by a particular buyer, then you find all the rules with right-hand side and out of those rules out of those rules you find I mean you sort these rules based on their confidence and suggest the items which are of very high confidence.

You can also make predictions and items can also be recommended, it appears in the RHS of the association rule supported by the active user. But anyway, top-M users cannot be predicted by using this kind of recommender system okay. Then the next one is your the next one is your demographic based recommender system which we will be seeing shortly. In this demographic based recommender system let me remind you so far we have used this rating data in case of collaborative filtering as well as association rule mining. In association rule mining how did we use the rating matrix? as such rating data we did not use but wherever the

ratings were given together, once again I remind you these ratings are not the explicitly given user rating, this may be generated by the system by observing the behaviour of the user, his browsing behaviour or some other how much time he staying and so on.

So in that matrix wherever the items were occurred together, that we consider for association rule mining, so association rule mining and collaborative filtering both use that rating matrix. Then content based filtering was using the item details that is the item vector, we had 3 things item vector and rating matrix and the user demographics, now this particular method we will be using user demographic data okay, let us see what happens in this.

(Refer Slide Time: 33:12)

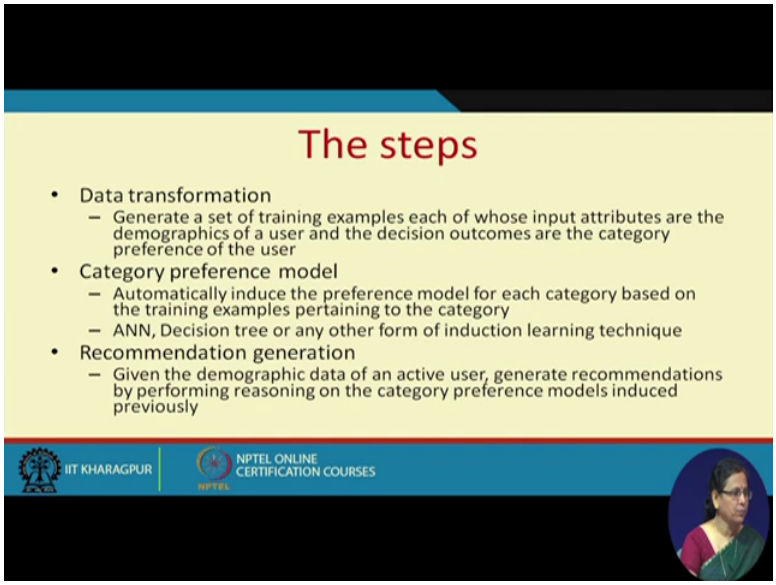


So in this approach , one has to recommend items to a user based on the preferences of the user whose demographics are similar to that of the user active user who is under consideration right now. Now unlike other approach where the recommendations are made at the item level, here the recommendations are made at the category level to help the user. So this is a more generalised form of information and this generalisation happens because of the sparsity problem, which means if 2 users are demographically similar that may not mean that all of them have seen similar items. So therefore there are maybe many sparse data there may be many elements in one user's rating vector which is not there in the second user.

Similarly something which is there in the second user may not be there in the first user. Now how this sparsity can be reduced? If we combine the items into groups it may so happened that one of the items of that group is is is commonly viewed, so that way it reduces the sparsing problem. So now what is the typical advertising for this application for this

particular recommender system? Basically it is used in for advertising, for targeted advertising purpose.

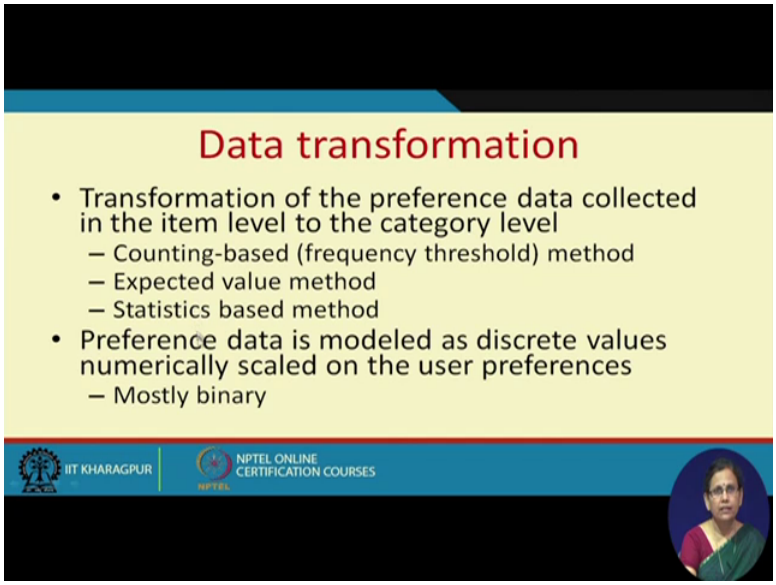
(Refer Slide Time: 35:02)



The steps here are the data transformation, during this stage you generate a set of training examples of training examples each of whose input attributes are the demographics of a user and the decision outcomes are the category preference of the user. So you have a predictive model in which the inputs and inputs are the items the user demographics then the output is the preference. Now, usually for this category to build for building this category preference model, and a generic predictive tool that is a artificial neural network, decision tree, et cetera can be used in fact, decision tree example we have already seen in content based so same thing can be used here as well.

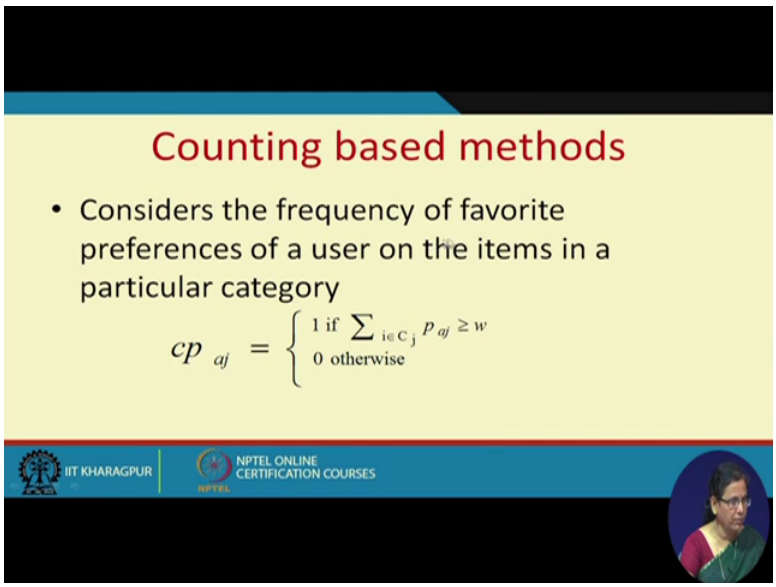
So then is the thing is the next you have to generate the recommendation. Now given the demographic data of an active user, one has to generate the recommendation by performing the reasoning on the category preference model which is nothing but a predictive model like artificial neural network, decision tree, et cetera, which is already trained on the demographic data.

(Refer Slide Time: 36:25)



Now in the data transformation stage, the transformation of the preference data which is collected at the item level to the category level is taken care off. Either you use see what is the input to the model? Input to the model is the user demographics, his age, his profession and so on. And what is the output? Output is the item at the category level. What is the category of the item? If I am buying a pen or a pencil or something which is related to writing or related to office stationery , all of them will be put together as one group okay.

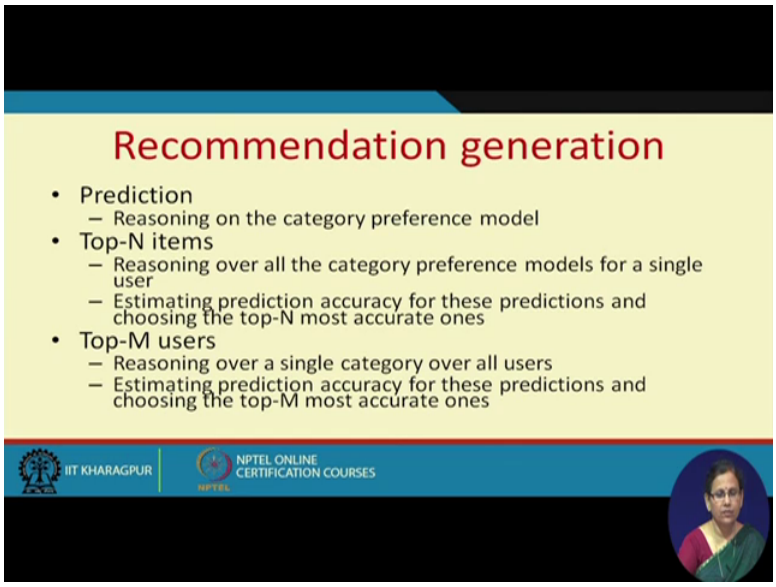
(Refer Slide Time: 37:38)



So now this preference data is modelled as discrete values numerically scaled on the user preference, they are mostly binary whether the user has seen item of a particular category or not seen the item. Now again the frequency of the favourite preferences of a user on an item

in a particular category can be 1 if it is greater than sum threshold value or it can it can be made as 0 so that is how you create the preference level, which is the output variable in a predictive model.

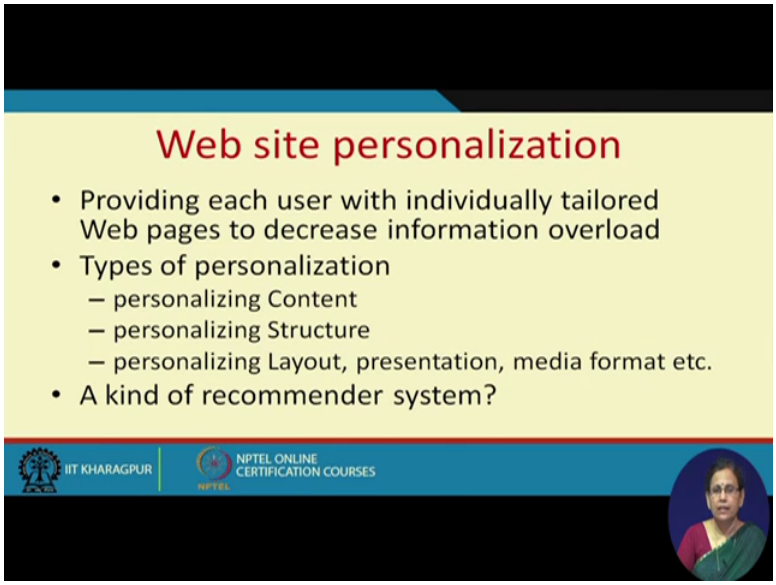
(Refer Slide Time: 38:06)



Then next is let us see how exactly once this model is built, how exactly recommendations are generated. Here the prediction, the category model is ready so when a user comes, the input is his demographic data and model is already trained with the past data and output is the category. Then in this you can also generate top-N items, how? You can reason over all the category preference models for a single user, when I say you have to reason over all the category preference models which means you have a model for each category. So out of those models out of those models you estimate the prediction accuracy I mean out of those models for a single user you find out those categories those categories where the score is very high.

So now here for estimating the prediction accuracy of these you choose the top-N most accurate ones then next come your Top-M users. Now over a single category for all the users, first one for top-N items for 1 user over all category preference model and second one for 1 category multiple users. So in this case you again find out you reason over all the category models and find out the top few once. Now you are estimating the prediction accuracy for this these things, you can which over are which so ever are of high values, you choose those accurate one.

(Refer Slide Time: 40:25)



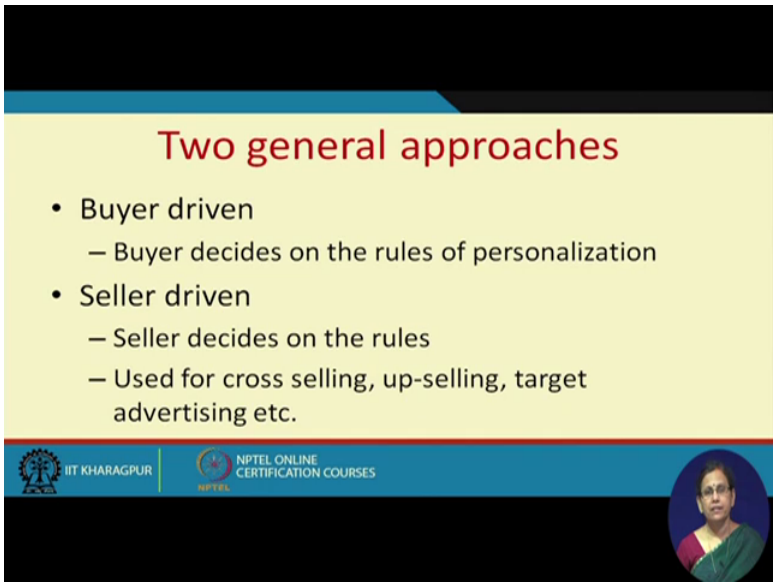
Now one very important application, recommender system you have already seen, in most of the commercial websites you see recommender systems whenever you make any products search. But besides this there is one very important application of recommender system is website personalisation. So this website personalisation uses recommender system approach for reorganising the web pages, so this personalisation can be at the content level, it can be at the structure level, it is about about personalising the layout, presentation, media format, et cetera.

(Refer Slide Time: 41:11)

Now, what is the advantage of using this personalisation? Look, this personalising can happen at 2 levels, one you give freedom to the user to personalise his page, he can decide

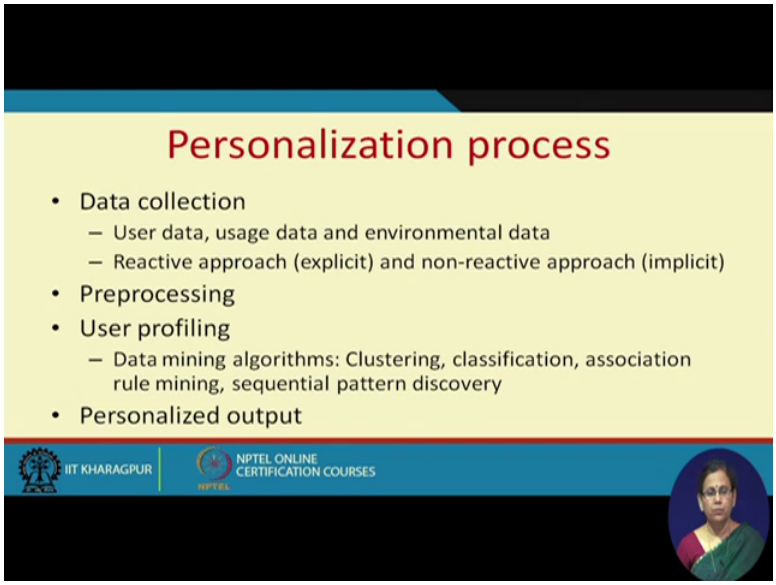
what to keep what not to keep. But if the process is automated, looking at his behaviour and matching his behaviour with behaviour of the other people and looking at his viewing pattern of different items, you can organise if one page is automatically organised, we can say it is the personalisation it is the automatic personalisation, it is not a situation where the client is given the freedom and what is the advantage? It increases site visibility, it converts users to buyers, it returns current customers and so on.

(Refer Slide Time: 42:06)



So this is what I was telling, it can be either buyer driven where the buyer decides the rules of personalisation and does it from his angle. Then it can be seller driven, where the seller decides on the rules and the seller arranges it looking at the generic behaviour of many users and use it for cross selling, up-selling, target advertising, et cetera.

(Refer Slide Time: 42:34)



So for the personalisation process, data can be collected at user level I mean that is the user's demographic data, it can be usage data, it can be environmental data. Then it can be either reactive approach where the data is collected explicitly or it can be non-reactive approach where the data is collected implicit manner. After the preprocessing, preprocessing et cetera is a generic step in every data centric applications then user profiling is made using many data mining algorithms like classification, et cetera, then finally from this model you generate a personalised output, thank you very much.