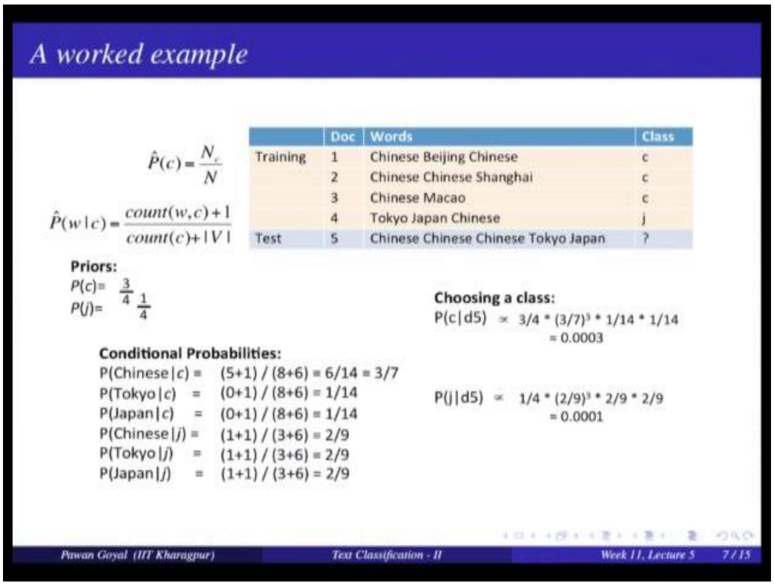
Natural Language Processing
Prof. Pawan Goyal
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 55
Text Classification – II

Welcome back for the final lecture of this week. So, we had started talking about text classification and we discussed the naive Bayes model for text classification and today we will take a working example for how do we use naive Bayes model and then we will discuss various other issues with classification and also talk about the evaluation of classification.

(Refer Slide Time: 00:41)

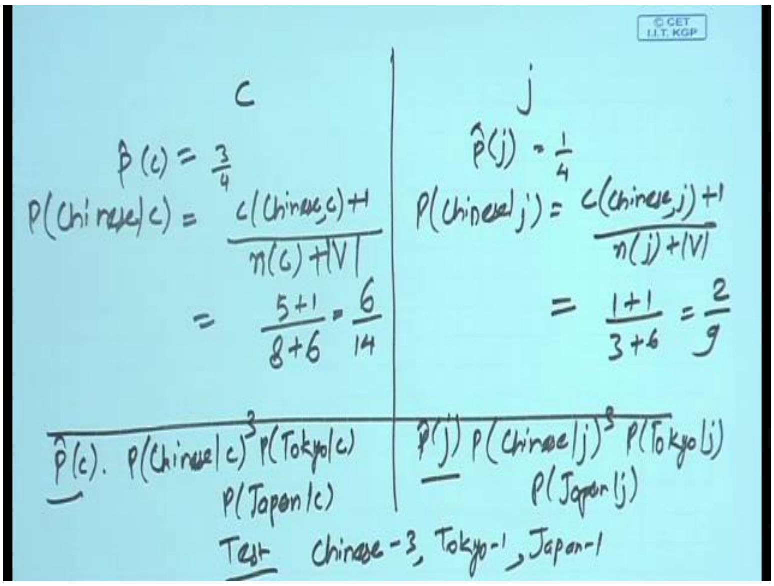


Let us take an example. So, what you are seeing here? So your training set that contains 4 documents document 1 to 4 and so document contains some words. So, like 1 contains Chinese Beijing Chinese and so on and there are some labels assigned. So, this document corresponds to class c that might be Chinese and this corresponds to class j that can be Japanese.

And then in test data you have 1 document; document number 5 that contains Chinese,

Chinese, Chinese, Tokyo, Japan and you want to assign it to one of the class; one of these 2 classes. So, how do we solve this problem using naive Bayes model? So, remember what is a naive Bayes model? We need these 2 probabilities P hat c and P hat w given c for all these 3 words. So, let us try to compute these probabilities.

(Refer Slide Time: 01:38)



I have class c i have class j. So, I need to find out what is P hat c and P hat j. So, what will be P hat c? That is the probability of class c in maintained data. So, it occurs 3 times out of 4. So, this will become 3 by 4 and this will become 1 by 4 then I need to compute different probabilities for words. So, what are the words here in my test documents? I have Chinese 3 times then Tokyo once, Japan once.

These are document in my test data; test document, I want to assign it to some class and what will be the probability of these 2 classes? The probability of class c would be P hat c times probability Chinese given c to the power 3 because it is agreeing 3 times probability Tokyo given c probability Japan given c and this will be P hat j probability Chinese given j to the power cube probability Tokyo given j and probability Japan given j. So, now, I already know this and this. So, I need to compute the other 3 probabilities. So, how do we compute probability Chinese given c? This would be number of times the word Chinese occurs with this class, count of Chinese with c and I am using (Refer Time:

03:41) smoothing. So, it will be plus 1 divide by all the words that occur in the class Chinese. So, I can call it number of words in class Chinese plus my vocabulary size.

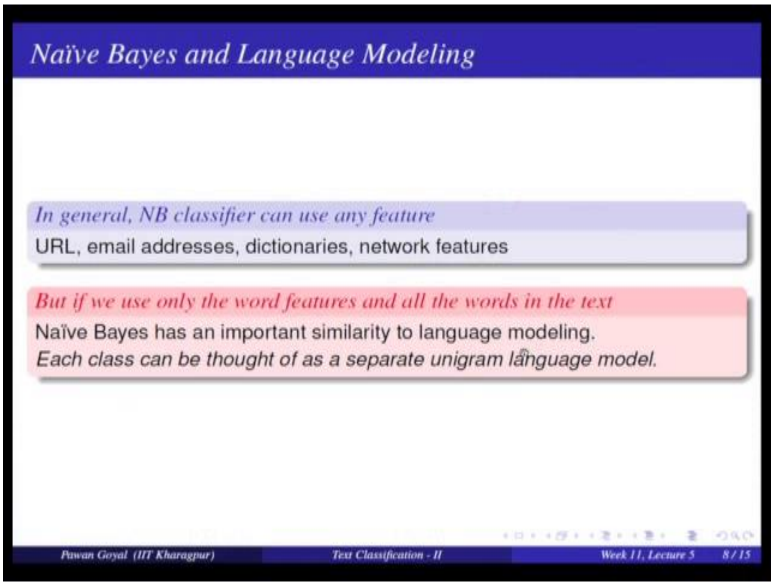
Now, what is the vocabulary size here and what is n c? Let us see, similarly you can compute probability Chinese given j as count of Chinese in class j plus 1 divide by number of words in class j plus my vocabulary size. So, let us see from the documents how many times the word Chinese occurs in class c. So, it occurs 1, 2, 3, times actually 1, 2, 3 and 4 times. In class j, it occurs once. What is the vocabulary size of Chinese 1 2 3 not vocabulary size, how many different tokens are there in class Chinese? 1, 2, 3, 4, 5, 6, 7, 8, in Japanese 1 2 3 and vocabulary size how many unique words? 1 Chinese, Beijing 2, Shanghai 3, Macao 4, Tokyo 5, Japan 6. So, I now know my variables. So, I have write it is this bit came out to be 5 plus 1 divide by number of words in class Chinese add for 8 vocabulary size was 6, this comes out to be 6 by 14, what about this? So, number of times Chinese occur in class Japanese was once plus 1, number of words in class Japanese were 3 plus 6. So, this comes out to be 2 by 9.

Similarly, now you will compute the other 2 probabilities that is probability Tokyo given c, Japan given c and Tokyo given j, Japan given j. So, let us see these on slides. So, once you know how to do that we can see that quickly on the slide. So, first you complete the priors that you know are 3 by 4, 1 by 4 then you complete all these condition probabilities. So, we already completed probability Chinese given c and Chinese given j, let us see the other 2. Tokyo given c would be number of times Tokyo occurs in class c 0 plus 1, 1 divide by 14. So, denominator will remain the same.

Tokyo given j will be 1 plus 1 2 divide by 9. So, in the Japan given c will be 1 by 14, Japan given j will be 2 by 9 and that is what you see 3 by 7, 1 by 14, 1 by 14, 2 by 9; 2 by 9, 2 by 9. Now can you compute the probability of class c given document 5? Probability of class c given document 5 that again comes out to be the same formula that we wrote and we have now all these values see you put all these values. So, this will be 3 by 4 times 3 by 7 to the power 3 times 1 by 14 times 1 by 14 and this will be 1 by 4 times 2 by 9 to the power 3 times 2 by 9 times 2 by 9.

And that will give you class c has a higher probability than class j. So, then this document d 5 will be assigned to class c and not to class j and this is how you use naive Bayes model its very simple to implement and you just need to compute the probabilities of different words given different classes and use that at 1 is smoothing.

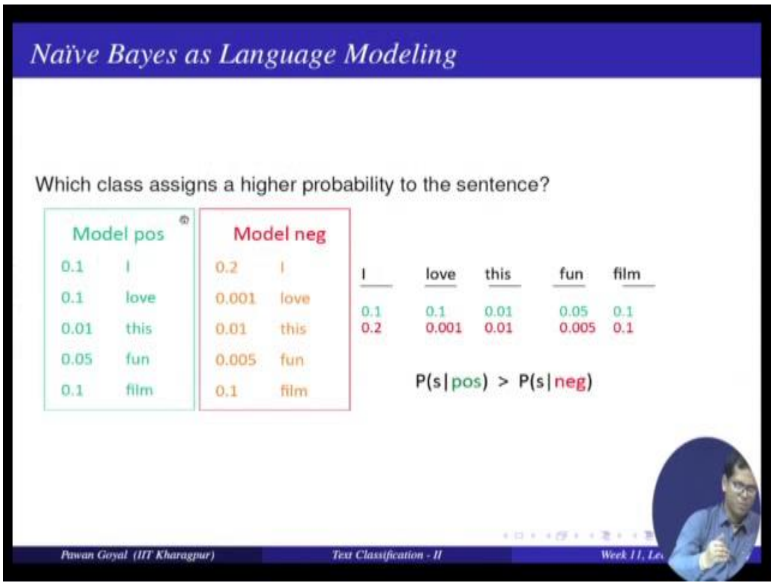
(Refer Slide Time: 07:32)



We saw, how do we use naive Bayes model of a classification now? So, there is one thing that I wanted to discuss is that how this is very close to the language model topic that we had discussed. So, how you can think of naive Bayes model as some sort of language model also. So, in general when you talk about naive Bayes model it is generic enough that can you (Refer Time: 07:58). So, you can use so if (Refer Time: 08:00) you can use what are the URLs, what are the email addresses, you can use some dictionaries, whether this word occurs in 1 of these dictionaries, what are the adverse features? For example, how many times you are getting emails from this person and so on.

Here you are allowed to use all these features inside a naive Bayes model, but suppose you are using only the content feature that is, what are all the words that are occurring in this text? So, if you use only the word features and all the words in the text then it has a very important similarity to the language modeling and what is that similarity if you think about it? So, it is like as if for different classes, you are having different language models. So, suppose a positive plus a negative plus and this is as if you are building language models for each of the class and then when you have a document at the test time, you are finding out which of this language model assigns a higher probability to this new document and that is the class of the document. So, this is an important similarity of language modeling and naive Bayes. So, you can think of each class as a separate unigram language model see; let us see some example.

(Refer Slide Time: 09:17)

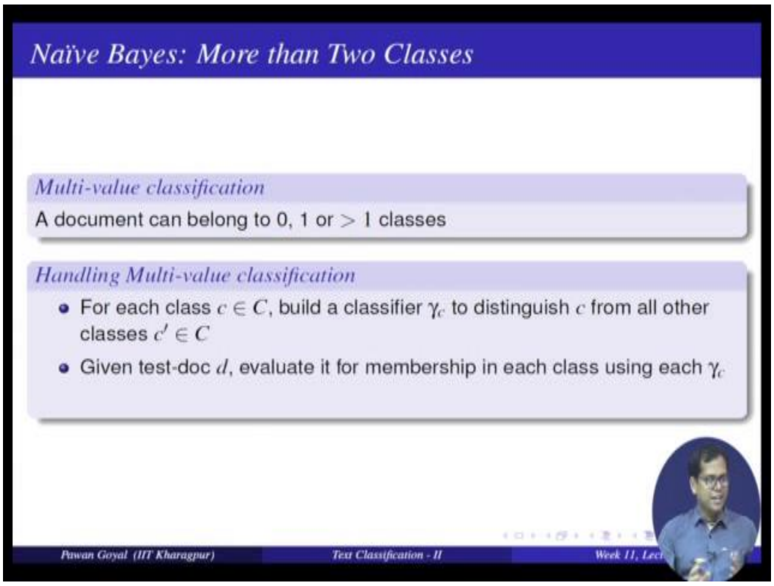


Like suppose you have 2 classes; positive and negative, now from training data you know what are the sentences that have labeled positive, you know sentences that have the label negative. So, take all the documents with the label positive edge 1 corpus and construct a language model out of that. Call it to a language model for the positive plus, similarly take all the words from the or all the documents from negative class, take it to a new corpus and build a language model unigram model, call it your negative model and now you find out the probability, you have the probability of different words as per these classes, suppose like your positive model gives a probability of 0.1 to I, 0.1 to love, 0.01 to this, 0.05 to fun and 0.1 to film. So, that is a in positive plus, you have a lot of times words like love, fun, etcetera coming. On the other hand, negative model you will not have these terms quite often. So, you will have I is occurring much more times, but love is occurring like point with a probability 0.001, fun with 0.005 and so on.

Now what will happen? At test time when you see a new document, so when you see a new document you will have a (Refer time: 10:36) you know all the words. Now try to assign this a probability as per both of these models. So, what is the probability for this sentence, I love this fun film as per my positive model and what is the probability for the sentence as per my negative model? And then you will see that immediately so positive model gives me a much higher probability then negative model and you can assign it to the positive plus and you can see that this is very much resembling, what you did in the naive Bayes model except that in naive Bayes model, you are also assigning a prior probability to each of these classes. So, here if both classes; they are roughly coming equal number of times in the data then this is like they are very much similar, either use language model or naive Bayes.

This is another way of thinking about this problem. So, think as if you can construct different language models for each of the classes and try to assign probabilities as per different language models. Now so, we talked about the case where there are 2 classes of multiple classes and a test document can belong to one of these classes only. So, you will use a naive Bayes model or any other model and find out what is the class that could be assigned to this test document.

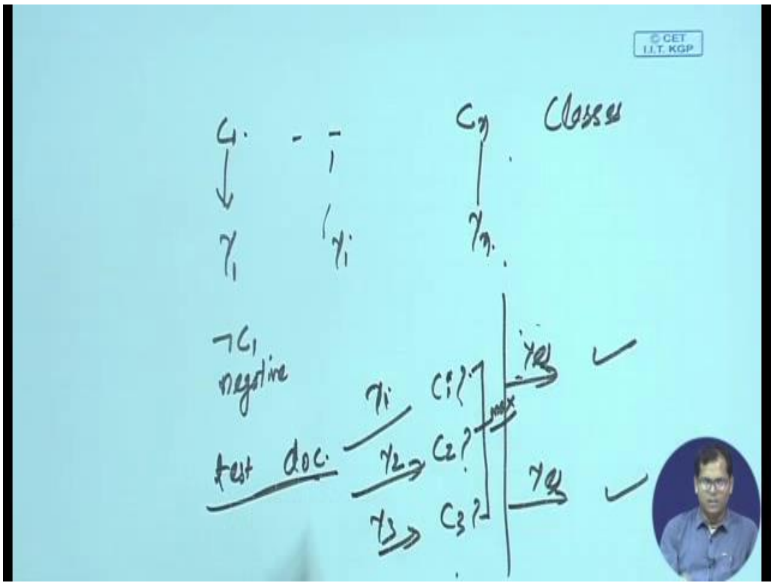
(Refer Slide Time: 11:56)



But suppose we are talking about a multi value classification problem, so what is a multi value classification problem? A document need not be assigned to 1 and only 1 class. So, it might happen that the document does not belong to any of the classes. So, it belongs to 0 classes in the set, it might have belongs to 1 class or it might belong to more than 1 classes also. So, that is where your categories are not mutually explosive. So, your document can belong to many of the categories at the same time.

Then how do you solve this problem using whatever we discussed, whatever technique we discussed. And a simple approach would be you make different binary classifiers for each of the classes. So, you have capital C classes, you have different binary classifiers for each of the classes and given a test document you try to assign a probability of this document to belong to each of the classes separately.

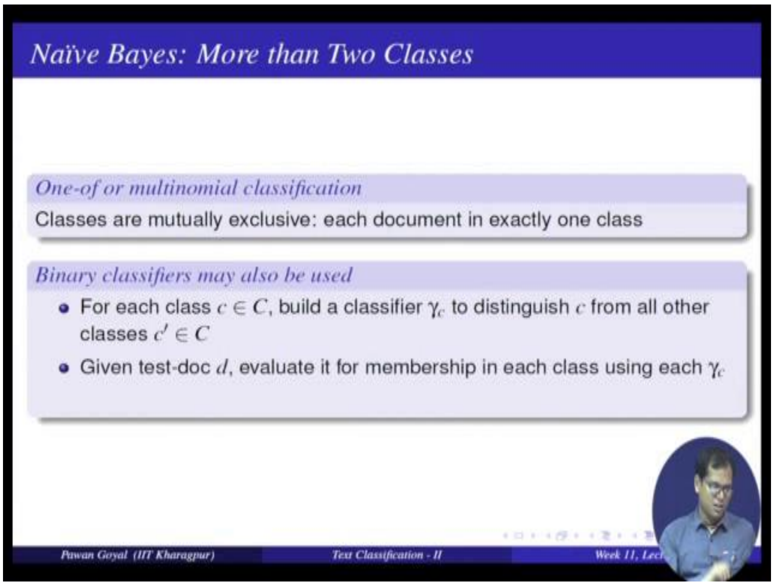
(Refer Slide Time: 12:56)



What we are saying? Suppose you have C 1 to C n; n classes so, what you will do? So you will take 1 class and build a binary classifier gamma 1, the document belongs to this class or does not belongs to this class, gamma i gamma n and different classifiers. How will you win this classifier? You need the data. So, C 1 will be the positive examples, what would be the negative examples? Everything that is not C 1, here the negative examples, so you can take any of these classes. So, similarly you can build all of these classifiers. Now once you build this classifier given a document a test time, test document; you whether know all these classifiers and find out whether it belongs to class C 1 C 2 C 3 and so on and whatever the classifiers says, yes, is the label given to the document. So, it might say yes in for 2 classes and it gets 2 labels.

The classifier might say no for each of the classes then it will not get any label. So, this in general, solves the problem of multi value classification where each document attach time might be given multiple different labels. So, what are we doing? So, for each class in my set, building a classifier gamma C that distinguishes this class C from all of the classes. Then given test document d you are evaluating its membership in each of these classes by the classifier gamma C and wherever gamma C returns to that is a label belonging to the document. So, document by this way can get many of the labels. Now this is a multi value classification.

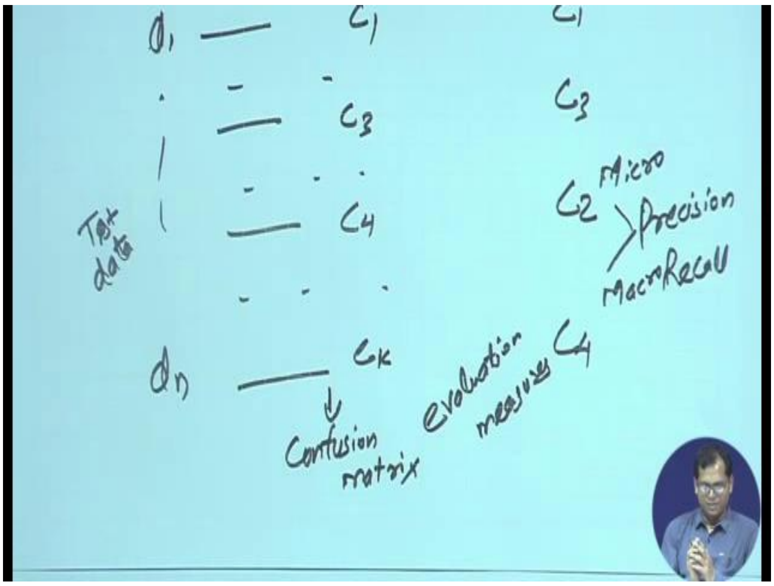
(Refer Slide Time: 14:53)



On the other hand you might have this 1 of a mutually exclusive classification where each document can belong to 1 and only 1 class. So, we can always handle this problem by defining multiple classes from a naive Bayes model, but suppose I am trying to build binary classifiers. So how do I handle this with binary classifiers? So, what you will do here? For each classifiers C, for each class C, you build a separate classifier like the way we did here. You build separate classifier for each of the classes. So, till here everything is same. So, each classes you are having separate classifiers test document you again run various classifiers like gamma 1 tell me whether C 1 is the; so now, instead of just saying yes and no, what you will say? You will say, what is the probability that this classifier gamma 1 is giving to class C 1?

What is the probability that gamma 2 is giving to C 2, what is the probability that gamma 3 is giving to C 3? We will get all these probabilities and whichever probabilities, the maximum you will assign that class to the test document if the class is are mutually exclusive. So, if they are not, simply take a decision when each of the cases if they are mutually exclusive, you will find a probabilities and take the max. So, that is another approach. So, given test document d you evaluate membership for each of the classes and d will belong to the class with the maximum score. So, that is how you can build multi class classification by using simple binary classifiers. So, now, let us talk about the evaluation part. How do we evaluate our text classification approach? So, in general assume that you have some n number of classes and so what do I mean by evaluation? So, you will have a training data to train your classifier, but there will be a test data where you will find out how good you classifier is predicting the classes.

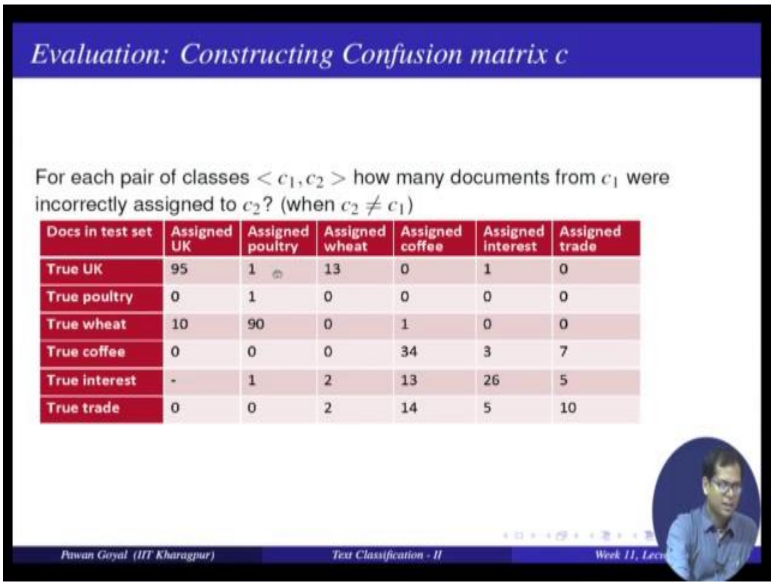
(Refer Slide Time: 17:14)



Attach data, what will happen? You will have some documents. So, d 1 to some d n and document into your test data, now you will run your classifier for each of these n documents and you will predict a class, suppose you say this belongs to class C 1, this belongs to class C 3, this belongs to class C 4, this belongs to class C k, whatever and like that for each document, you are assigning some class separated class, now how do you know how good your model is? You have to compare it with the gold standard or the ground truth. So, then you can say that what is the true class? Suppose this is actually C 1, this is C 3, this is C 2, this is C 4 and so on. This is the true class and even to match the true class with the predicted class. So, what are the different evaluation measures that can be used to compare this with this why is the simple accuracy what fraction of times your classifier gives the same answer as that ground truth that is one thing this is not very popular there are some other measures like precision recall and in precision there are 2 evaluations micro and macro.

Let us see how do we compute all these different evaluation measures? For doing that the first thing you need to do is to convert this whole thing into a confusion matrix. So, what is the confusion matrix in confusion matrix? So, what do you will have for each class how many labels were there in the predicted in the in the true class and how many you could predict. So, it will be like a. So, it will be what how many you got it correct how many you not get it correct everything will be there in that confusion matrix.

(Refer Slide Time: 19:26)



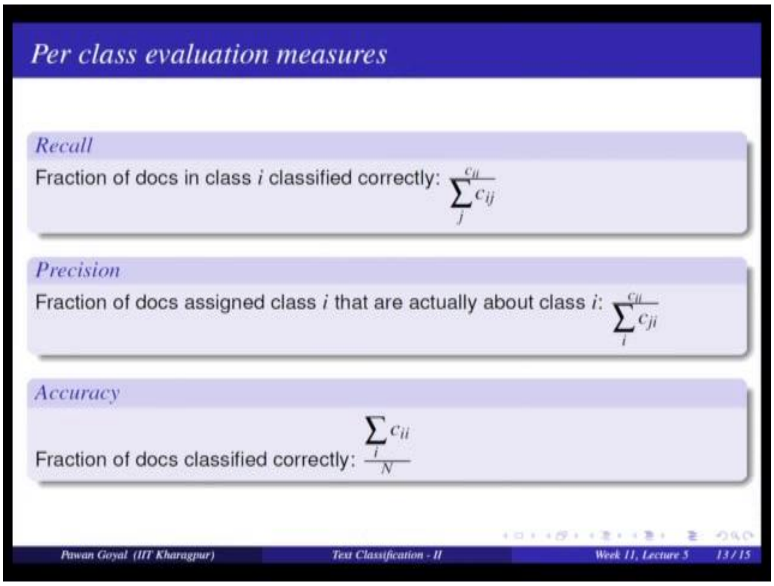
Let us see 1 example. So, suppose I have here 6 classes; UK, poultry, wheat, coffee, interest and trade. So, you are seeing the rows are true classes; that means how many documents in these classes and this is assigned classes. So, now, how do you read this matrix? So, what this number and 95 means that you classifier assigned 95 documents to UK and they were true, but this whole column corresponds to whatever your classifier assigned a UK. So, the assigned 10 documents to class UK while they were from class wheat. So, immediately you can see that my classifier has some confusion between the classes UK and wheat. So, that is why this is called the confusion matrix, where is my classifier confusion? So, 95 cases, it assign UK, it was actually UK 10 cases is assigned wheat, sorry UK, but it was wheat. So, let us take another 1 in poultry, what is happening? 1 case where it assign poultry, what the actual class was UK, 1 case, it has assigned poultry into actually poultry 90 cases, it assigned poultry, but it was actually wheat so; that means, the classifier is really confused in that it is assigning poultry to 90 documents that were actually wheat and similarly you can read all your confusion matrix.

Now from this confusion matrix, suppose I ask you simple question like how many documents in the test data that classifier assigned to class UK and you can simply add all these, say 95 plus 10; 105, how many did it assigned to poultry? 1 plus 2; 1 plus 1, 2, 92, 93, so on, how many documents were actually belonging to class UK? That is why you have to read the row corresponded to UK, 95, 96, 109, 110; 110 documents belong to class UK.

So, now, suppose you have this confusion matrix, you can find out where your classifier is getting confusion all that and you can try to define your classifier with this information, but now suppose you have this final confusion matrix and you want to now see what is the precision of my classifier. So, let us see we tried to find out precision; individual precision. So, what is the precision of the class UK? By precision, I mean whatever documents my classifier assigning as UK, what fraction of them are actually UK? So, let us see, it is assigning 105 documents to UK, 95 of them are actually UK, so, precision for class UK would be 95 divided by 105.

Now, what would be the recall for the class UK? Recall means among whatever documents that are in the class UK, what fraction of the documents could my classifier accurately classify? So, here they were 110 documents in the class UK, my classifier could accurately classify 95 of those. So, recall would be 95 divided by 110, like that you can compute for each of the classes the individual precision and recall.

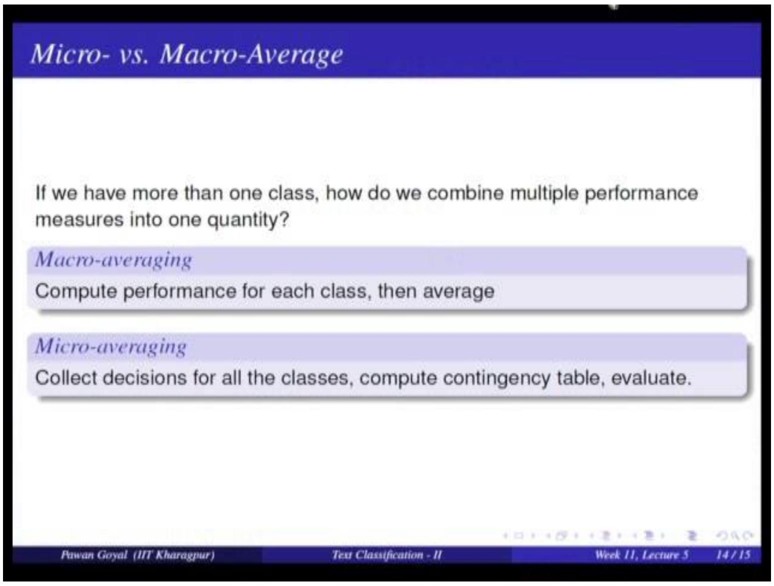
(Refer Slide Time: 22:47)



Recall would be whatever fraction of documents in class i corrected classified correctly and that you can see you can get by adding overall the all the elements in the row like that is what we did for the class UK added all the elements in the row C ii divided by summation over C ij for all j and precision fraction of documents that are assigned to class i that are actually about class a. So, this will be again C ii divided by now you will add the whole call. So, summation over i C ji and accuracy would be add all your diagonal elements divided by the total number of test documents that would be your accuracy.

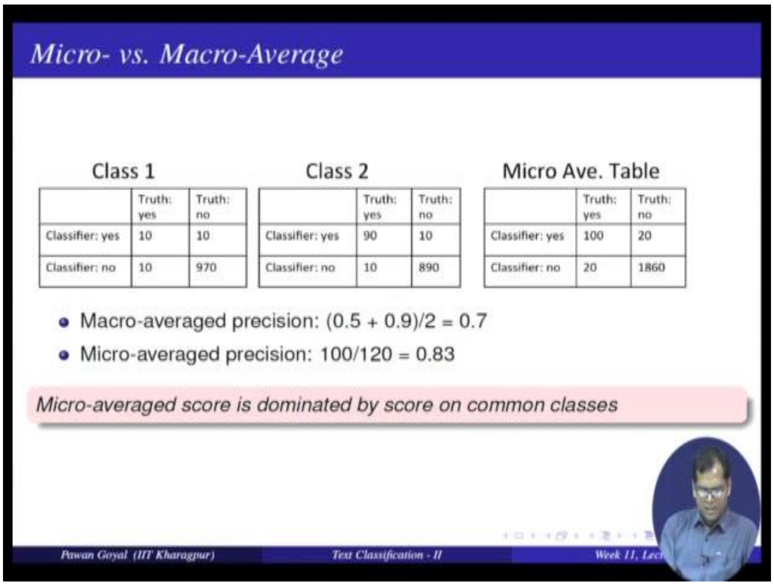
Now, we say that we can also compute the. So, this is for each and individual class and accuracy you can you need to talk about the overall test data, but can you compute precision for. So, micro and macro average precision for this test data.

(Refer Slide Time: 23:58)



What is this? So, if we have more than one class then how do we combine the performance measures for individual classes to make a single evaluation measure that is my micro and macro average precision? So, what is the difference? So, macro average what I will do? I will compute the precision or accuracy precision for is in the individual class and i will taken average over each suppose my class one is having a precision of 0.7 class 2 is having a precision of 0.9 I will take an average 0.7 plus 0.9 divide by 2.8 in micro average precision what I will do? I will first come take all the decisions and compute a single matrix over all the decision over taken together. So, I will see how many cases are taking as true, false, I will take them together make a single statistic or single condition suitable and I will compute my precision recall from that.

(Refer Slide Time: 25:01)



Let us take an example suppose for the class one there were 10 cases where you classifier said yes and the actual answer was also yes 10 cases where classifier said yes, but they were not belonging to the class 1, 10 cases where your classifier said no and they were actually belonging to the class one and are the 970 cases where classifier said no and they were not belonging to class 1 and this is one of the way of writing the confusion matrix for contingency suitable for each class. So, that is classifier yes no; truth yes no.

From there can you compute, what is the precision for class 1? So, it classify assign 20 documents to class one out of them 10 were correct precision is 0.5. Now let us see the second class, in the same manner we constructed the contingency table. So, what is the precision for class 290 were assigned directly out of 100 given by the classifier. So, it is 0.9.

So, now, when we have to do micro average precision we combine all these statistics. So, we combine these statistics say they were 10 plus 90, 100 cases where classifier said yes and this was also yes 10 plus 10; 20 cases were classifier said yes, but it was not the correct class similarly 20 cases here 160 cases here. So, this is you combine all these statistics and do a single table that becomes your micro average table. Now from once you have this table, how do you compute your macro average precision and macro average precision. So, micro average precision you compute precision for class 1, class 2 taken average. So, you have already computed this is 0.5 and this is 0.9 taken average. So, it becomes 0.7. So, this is your macro average precision.

Now, what to do a macro average precision? So, you will compute precision over this table now. So, that will be 100 divided by 120. So, that is point roughly 0.83. So, micro average precision comes out to be 0.7 and micro average comes out to be 0.83. So, now, what do you see o. So, why micro is coming out to be higher than macro precision? So, because in macro precision, you are giving in equal weight to all the classes, you are computing precision over class 1, classes 2 then you forget how many instances where there for each of the class. So, and they are given equal weight and you compute an average precision in micro average precision what is happening the class that is having more number of instances getting a higher weightage that is why it is bias towards class with classes that are having higher number of instances.

If you act if you are building a classifier of a multiple classes and you want to see that all your class if classes perform well then macro average precision is a good precision, if your classes are imbalanced, some classes are having high number of data points and other. So, macro average score is dominated by the score on the common classes. So, when you are talking dealing with test classification they might be some issues like in your training data some classes are more common than others. So, here more samples from some classes than other classes and this sometimes Bayes your classifier also. So, good strategy is that your sample, the number of instances in the classes.

So, one simple thing is you can do under sampling. So, if a classes very common you under sample it such that it becomes close to the other classes that are quite rare. So, that by that way in you are training data you will have roughly equal amount of samples in each of the classes and there are other algorithms that also allow you to over sample some of the minority class and so on. So, we will not discuss that in detail just wanted to tell that it might happen that you have an unbalanced data set then you have to do certain strategies before applying the classifier.

So, that finishes our week eleven for this course and then. So, next week we will discuss in detail about the sentiment analysis and opinion in next lecture. So, this will be topic for the final week.

Thank you.